

# NAVAL POSTGRADUATE SCHOOL MONTEREY, CALIFORNIA



## THESIS

### A STUDY OF HELICOPTER ROTOR DYNAMICS AND MODELING METHODS

by

Daniel S. Hiatt

September, 1995

Thesis Advisor:

E. Roberts Wood

Approved for public release; distribution is unlimited.

19960220 014

DTIC QUALITY INSPECTED 1

# REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE September 1995		3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. A STUDY OF HELICOPTER ROTOR DYNAMICS AND MODELING METHODS				5. FUNDING NUMBERS	
6. AUTHOR(S) Daniel S. Hiatt					
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey CA 93943-5000				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.					
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.				12b. DISTRIBUTION CODE	
<p>13. ABSTRACT (maximum 200 words)</p> <p>The rotor system is the primary source of vibratory forces on a helicopter. Vibratory forces result from the rotor system response to dynamic and aerodynamic loading. This thesis discusses sources of excitation, and investigates rotor system modeling methods. Computer models based on finite element and Myklestad methods are developed and compared for the free and forced vibration cases of a uniform rotor blade. The modeling assumptions and the effects of non-uniform physical parameters are discussed. The Myklestad based computer model is expanded to include coupling effects inherent in modern rotor blades. This rotor modeling program is incorporated into the Dynamics section of the Joint Army/Navy Rotorcraft Analysis and Design (JANRAD) program currently used by the Naval Postgraduate School's helicopter design course (AA4306) for preliminary helicopter design and analysis.</p> <p>Computer programs are developed as tools to investigate the stability of a rotor system for the specific cases of rotor flapping and ground/air resonance. A rotor flapping stability model, based upon Floquet theory, provides a means of analyzing the effect of increasing advance ratio on the flapping stability of a rotor system. The ground/air resonance model uses a constant coefficient approximation of the rotor system to allow analysis of the effects of coupling between rotor lag motion and airframe motion.</p>					
14. SUBJECT TERMS Helicopter, rotor, dynamics, finite element, exact method, Myklestad, Floquet theory, ground resonance				15. NUMBER OF PAGES 134	
				16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL		

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)  
Prescribed by ANSI Std. Z39-18 298-102



Approved for public release; distribution is unlimited.

**A STUDY OF HELICOPTER ROTOR DYNAMICS AND MODELING  
METHODS**

Daniel S. Hiatt  
Lieutenant, United States Navy  
B.M.E., University of Minnesota, 1985

Submitted in partial fulfillment  
of the requirements for the degree of

**MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING**

from the

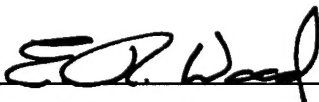
**NAVAL POSTGRADUATE SCHOOL  
September 1995**

Author:



Daniel S. Hiatt

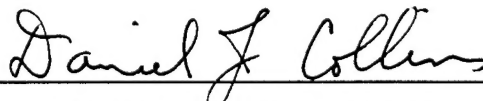
Approved by:



E. Roberts Wood, Thesis Advisor



Jon D. Raggett, Second Reader



Daniel J. Collins, Chairman

Department of Aeronautics and Astronautics





## ABSTRACT

The rotor system is the primary source of vibratory forces on a helicopter. vibratory forces result from the rotor system response to dynamic and aerodynamic loading. This thesis discusses sources of excitation, and investigates rotor system modeling methods. Computer models based on finite element and Myklestad methods are developed and compared for the free and forced vibration cases of a uniform rotor blade. Modeling assumptions and the effects of non-uniform physical parameters are discussed. The Myklestad based computer model is expanded to include coupling effects inherent in modern rotor blades. This rotor modeling program is incorporated into the Dynamics section of the Joint Army/Navy Rotorcraft Analysis and Design (JANRAD) program currently used by the Naval Postgraduate School's helicopter design course (AA 4306) for preliminary helicopter design and analysis.

Computer programs are developed as tools to investigate the stability of a rotor system for the specific cases of rotor flapping and ground/air resonance. A rotor flapping stability model, based upon Floquet theory, provides a means of analyzing the effect of increasing advance ratio on rotor system flapping stability. A constant coefficient approximation of the rotor system allows analysis of the effects of lag motion and airframe motion coupling in ground/air resonance.



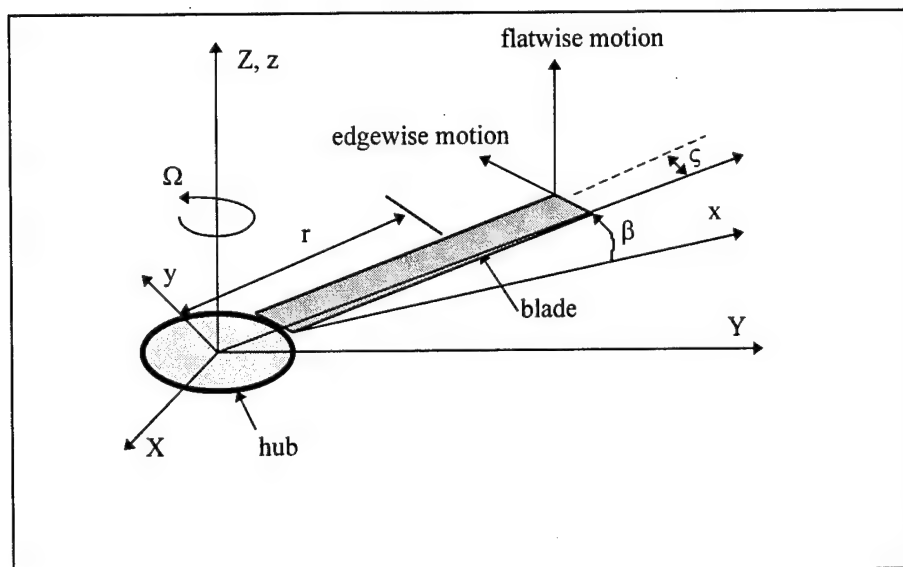
## TABLE OF CONTENTS

I. INTRODUCTION.....	1
A. THE PHYSICAL ROTOR.....	1
B. THE OPERATING ENVIRONMENT.....	1
1. Random Response.....	2
2. Steady-State Response.....	2
3. Self-Excited Vibrations.....	4
C. ROTOR SYSTEM MODELING AND ANALYSIS PROCEDURES.....	5
II. ROTOR BLADE STATIC AND DYNAMIC FREE RESPONSE.....	7
A. ROTOR BLADE MODELING METHODS.....	7
1. Exact Method.....	7
2. Myklestad's Method.....	8
a. Concept.....	8
b. Software.....	9
3. Finite Element Method.....	11
a. Concept.....	11
b. Software.....	12
B. ROTOR BLADE MODEL COMPARISON.....	14
1. Non-rotating Uniform Blade Case.....	14
2. Rotating Uniform Blade Case.....	17
3. The Effects of Non-uniformity.....	21
III. ROTOR BLADE DYNAMIC FORCED RESPONSE.....	23
A. FORCED RESPONSE OF THE ROTOR BLADE.....	23
B. FORCED RESPONSE USING THE FINITE ELEMENT METHOD.....	24
C. FORCED RESPONSE USING THE MYKLESTAD METHOD.....	27
D. JANRAD FORCED RESPONSE MODEL.....	28
1. Rotor Blade Model Input Requirements.....	28
2. Rotor Blade Model Calculations.....	29

3. Rotor Blade Model Outputs Available.....	30
IV. ROTOR SYSTEM FORCED RESPONSE AND STABILITY.....	39
A. ROTOR FLAPPING STABILITY.....	39
1. Rotor Blade Flapping Equation of Motion.....	39
2. Rotor flapping Stability Model Software.....	41
3. Software Output.....	42
B. AIR AND GROUND RESONANCE STABILITY.....	46
1. Rotor System Equations of Motion.....	46
2. Rotor Lag Stability Model Software.....	47
3. Software Outputs.....	48
V. CONCLUSIONS AND RECOMMENDATIONS.....	53
A. CONCLUSIONS.....	53
B. RECOMMENDATIONS.....	53
APPENDIX A. BEAM BENDING VIBRATION BY EXACT METHOD.....	55
APPENDIX B. BEAM BENDING VIBRATION BY MYKLESTAD'S METHOD.....	59
APPENDIX C. BEAM BENDING VIBRATION BY FINITE ELEMENT METHOD..	63
APPENDIX D. FLOQUET THEORY.....	69
APPENDIX E. BLADE LAG EQUATIONS OF MOTION.....	71
APPENDIX F. COMPUTER PROGRAMS.....	75
A. DYNAM.M.....	76
B. MYKBIS.M.....	86
C. BLDFAN.M.....	88
D. ROTOR.M.....	91
E. RTRTOT.M.....	93
F. BLDREV.M.....	99
G. OUTPUT.M.....	104
H. ADVFLOQ.M.....	110
I. COEFF.M.....	112

J. CCGRES.M.....	113
LIST OF REFERENCES.....	115
INITIAL DISTRIBUTION LIST.....	117

## LIST OF SYMBOLS



The figure above presents the conventions used in this paper with respect to the rotor system, unless otherwise noted. The following notation applies unless otherwise noted:

- $c_i$  damping coefficient of the  $i$ th blade
- $c_x$  hub effective damping coefficient in the  $x$ -direction
- $c_y$  hub effective damping coefficient in the  $y$ -direction
- $c$  rotor blade chord
- $C_n$  flatwise aerodynamic damping force
- $C_\theta$  lag damping coefficient
- $D_n + j d_n$  aerodynamic drag force acting on  $n$ th blade element for a particular frequency component  
 $D_n \sin \omega t + d_n \cos \omega t$
- $dC_l / d\alpha$  lift curve slope for the blade airfoil
- $dm$  incremental mass
- $dT$  incremental thrust force
- $e$  rotor blade flap or lag hinge offset - measured from rotor center of rotation to hinge center
- $E$  Young's modulus
- $f$  applied nodal force
- $f(x, t)$  time dependant distributed force
- $F_n + j f_n$  aerodynamic lift force acting on  $n$ th blade element for a particular frequency component

	$F_n \sin \omega t + f_n \cos \omega t$
g	deflection about flatwise principal axis due to applied load
G	deflection about chordwise principal axis due to applied load
h	length of element
I	rotor blade cross-sectional area moment of inertia
$I_b$	rotor blade second mass moment about lag hinge
j	$\sqrt{-1}$ unless used as an index
$k_i$	lag hinge spring constant for ith blade
$k_x$	hub effective spring constant in x-direction
$k_y$	hub effective spring constant in y-direction
$K_p$	flap proportional feedback gain
$K_R$	flap rate feedback gain
L	length of beam
m	mass per unit length
$m_b$	mass of rotor blade
$m_n$	mass of the nth blade element
$m_x$	effective hub mass in x-direction of motion
$m_y$	effective hub mass in y-direction of motion
M	moment ( blade aerodynamic moment coefficient with subscript)
n	integer number, i.e. number of rotor blades
$n_\psi$	number of azimuth angles
r	radius along the blade measured from the center of rotation
R	blade total radius
S	shear force
$S_b$	rotor blade first mass moment about lag hinge
t	time
T	tension or period as defined
u	slope measured from flatwise principal axis due to applied load
U	slope measured from chordwise principal axis due to applied load
v	slope measured from flatwise principal axis due to applied moment
V	slope measured from chordwise principal axis due to applied moment
$V_c$	centrifugal strain energy
w	displacement in an element
X	edgewise blade motion in absolute frame



$Z$	flatwise blade motion in the absolute frame
$\alpha$	rotor blade angle of attack, positive for leading edge up
$\beta$	blade flap angle
$\beta_0$	blade steady coning angle
$\phi(t, t_0)$	floquet state transition matrix
$\zeta$	blade lag angle, positive counter clockwise
$\partial$	partial derivative
$\Omega$	rotor speed
$\Lambda$	matrix of eigenvalues
$\gamma$	rotor blade Lock number
$\lambda$	individual eigenvalue
$\mu$	rotor advance ratio, $V_{\text{blade tip}}(\cos\alpha)/\Omega R$
$\nu$	rotor blade flapping natural frequency
$\theta$	slope of rotor segment in absolute frame or nodal slope
$\rho$	air density
$\sigma_x$	axial stress
$\omega$	excitation frequency
$\omega_n$	natural frequency of vibration
$[\omega_n]$	matrix of natural frequencies
$\psi$	rotor blade azimuth angle, measured counterclockwise from the blade aft position
$\xi$	$1-x/h$
$[c]$	element damping matrix
$[C]$	global damping matrix
$[D]$	dynamical matrix
$[I]$	identity matrix
$[k]$	element elastic stiffness matrix
$[k_c]$	element centrifugal stiffness matrix
$[K_c]$	global centrifugal stiffness matrix
$[K_E]$	global elastic stiffness matrix
$[m]$	element mass matrix
$[M]$	global mass matrix
$\{p\}$	element applied force vector
$\{P\}$	global applied force vector
$\{q\}$	displacement in modal coordinates

$\{u\}$	eigenvector
$[U]$	nodal to modal transformation matrix
$\{w\}$	displacement in nodal coordinate frame

## SUBSCRIPTS

$c$	center of mass
$h$	hub reference
$i$	ith component, increment or station
$n$	nth component, increment or station
$x$	x direction
$y$	y direction
$\psi$	related to azimuth
$0$	steady state or zero condition
$\infty$	free stream value
$\beta$	blade flapping angle
$\beta'$	$d/dt$ of blade flapping angle
$\theta$	blade pitch angle, positive nose up

## SUPERSCRIPTS

$E$	edgewise component
$F$	flatwise component
$0$	steady state
$\cdot$	$d/dt$
$\ddot{\phantom{x}}$	$d^2/dt^2$
$\rightarrow$	vector
$—$	rotating coordinate system
$*$	modal coordinate system



## ACKNOWLEDGEMENT

This thesis represents the work of unbounded curiosity. It is dedicated to my grandparents, Crawford and Elsie Ford... who sacrificed for my education; my parents, C. F. and Carolynn Hiatt... who led me by example; my siblings, Cliff, Debbie, and David... who had more confidence in me than I did; and, most importantly, my wife, Tracey, and my sons, Garrett and Ross... who made sure I always kept one eye on what was important.

I would also like to thank Professor E. Roberts Wood for keeping me honest and Professor Jon D. Raggett for a healthy dose of reality.

# **I. INTRODUCTION**

## **A. THE PHYSICAL ROTOR**

Any discussion on the dynamics of helicopter rotor systems must start by acknowledging that the complexity of their construction and the interaction between their components causes the most difficulties for dynamic analysis. Rotor blades are fabricated from a variety of materials, each with its own set of physical properties. The blades twist and taper to varying degrees and often incorporate different airfoil shapes over the length of the blade. All these factors result in rotor blades that are non-uniform in their properties and highly complex in their dynamics. As if these rotor blades were not difficult enough to study, they attach by various methods to rotor hubs of differing designs, materials and dynamics. The two most common forms of rotor blade to hub attachment, rigid and articulated, impose their own constraints on the motion of the rotor blade. Since the blades tie to a central hub, the motion of each blade as it rotates through space influences not only the hub to which it attaches, but also the other blades and vice versa.

It is not possible to model accurately all aspects of the physical rotor system nor is it necessary in most cases. If the goal of the dynamic analysis is preliminary design or qualitative behavior, a set of simplifying assumptions can be applied to the physical model to make the analysis tenable. Of prime interest is the effect of the out-of-plane (flapping) motion and in-plane (lag) motion of the rotor blades, and combinations of both. This paper discusses some of the methods for modeling these effects on the rotor and the assumptions necessary to implement them. To this end, the discussion is limited to free and forced dynamic response of the rotor system and to its stability in pure flapping and pure lag motion.

Several software programs written in MATLAB version 4.2 code provide the tools needed for rotor system dynamic analysis. These tools were developed by the author to expand the Joint Army/Navy Rotorcraft Analysis and Design (JANRAD) software program currently in use by the Naval Postgraduate School's helicopter design course (AA4306). The programs determine the natural frequencies of a rotor system and the system response to inflight loading based on the rotor physical model. They also provide a means for the qualitative study of the dynamic flapping stability of a rotor system in flight and its tendency to enter resonance on the ground and in the air.

## **B. THE OPERATING ENVIRONMENT**

The rotor is subjected to a continuously changing environment comprised of aerodynamic forces in conjunction with control input loads and accelerations. In essence, modeling the rotor's operating environment is as challenging as modeling the physical rotor. Once again the use of simplifying

assumptions can lead to an acceptable simulation of the actual environment. If the forces applied to the rotor are divided into the categories of steady, random and self-exciting vibration forces, the dynamic response of the rotor can be more easily analyzed.

## 1. Random Response

Random forces are those that are not periodic in nature, such as control input transients and random gusts along with wake interaction between rotor blades. These forces are unpredictable and extremely difficult to model accurately. They are important because they excite the rotor system in its fundamental, or natural, modes where the total response of the rotor is given by the superposition of all the individual modes. Knowledge of the rotor system natural frequencies and the dynamic magnification of the excitation force is critical to the analysis of the rotor dynamic response. The determination of rotor system natural frequencies is covered by this paper, but random excitation force modeling and application are not.

## 2. Steady-State Response

The term "steady-state force" does not imply that the force is constant valued, but rather that it represents the steady state of the forces applied to the rotor system in flight; free of all random and self-exciting vibration forces. In reality this steady-state force is periodic in nature and is the superposition of individual force components at integral multiples of the rotor rotational speed (called  $n/rev$  where  $n$  is an integer corresponding to the number of occurrences per revolution of the blade). One key assumption in the application of a steady-state force is that each blade is subjected to the same loading as it passes the same azimuth in rotation. An example of a typical steady-state inflight thrust force generated by a rotor blade near its tip is illustrated in **Figure 1-1**.

Figure 1-1 depicts the thrust force generated by the rotor blade through one revolution as a closed cycle where it always returns to the original starting value. The assumption that the aerodynamic forces are periodic in nature means they can be decomposed into a constant valued component and an infinite set of purely harmonic components, each one at an integral multiple of rotor rotational speed. The classical means of decomposing the thrust force into harmonic components is through Fourier analysis where:

$$F(\psi, r) = F_0(r) + \sum_i F_i(r) \sin(i\psi) + j f_i(r) \cos(i\psi)$$

**Equation 1-1**

is an expression of thrust force in complex form such that

$$F_0(r) = \frac{1}{n_\psi} \sum_{j=1}^{n_\psi} dT(\psi, r)$$

**Equation 1-2**

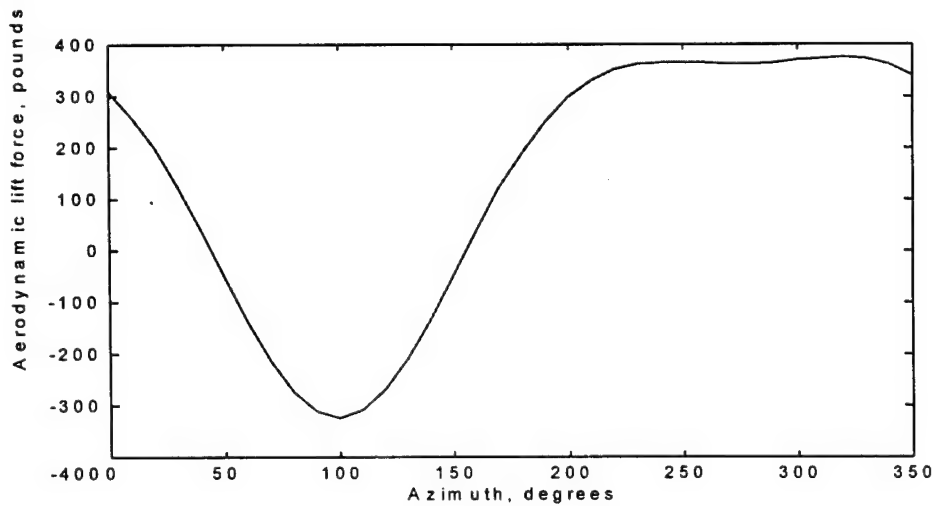
represents the steady aerodynamic thrust component and

$$F_i(r) = \frac{2}{n_\psi} \sum_{j=1}^{n_\psi} dT(\psi, r) \sin(i\psi) \quad i = 1, 2, \dots, \infty$$

$$f_i(r) = \frac{2}{n_\psi} \sum_{j=1}^{n_\psi} dT(\psi, r) \cos(i\psi)$$

**Equation 1-3**

represents the  $i$ th harmonic steady-state thrust components. In reality, an infinite set of components is not required since the magnitude of the higher harmonic components is negligible. This phenomenon is illustrated by performing Fourier analysis on the thrust force depicted in **Figure 1-1**. The steady and first 6 harmonic sine components of the Fourier analysis are plotted as **Figure 1-2**. **Figure 1-2** shows the magnitude of the thrust force decreases rapidly with each increase in harmonic number. At some point the harmonic component becomes too small to influence the dynamics of the rotor. In general practice, only the steady and first ten harmonic components of the thrust force are used to simulate the actual force applied.



**Figure 1-1 Typical aerodynamic thrust force developed by a rotor blade in forward flight.**

Modeling the applied forces as steady and harmonic components is convenient because the solution of the forced rotor equation of motion will also be harmonic in nature. The magnitude of the rotor dynamic response is directly related to the magnitude of the applied forces, to the system damping and relationship of the forcing frequencies to the rotor system natural frequencies. Just as the applied force is assembled from steady and harmonic components, the total response of the rotor system is assembled from superposition of the blade response to each individual harmonic of applied force.

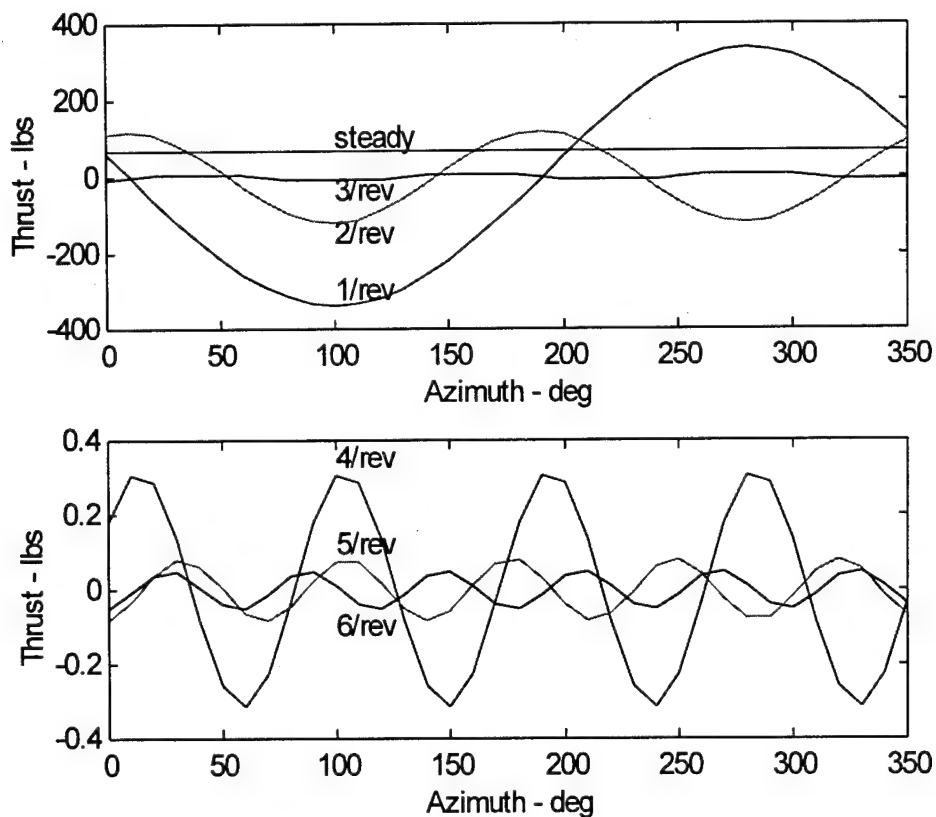


Figure 1-2 Results of Fourier analysis on thrust data depicted in Figure 1-1

### 3. Self-Excited Vibrations

Self-excited vibration response is related most closely to the stability or damping of the rotor system. The amount of damping governs how the system responds to a set of operating conditions. In the case of negative damping, the energy needed to drive oscillations to grow without bound is not generated by the rotor system, but requires an external source of energy. For classical ground resonance, this external



source of energy is provided by the engine. Generally these conditions require that the rotor system damping is negative or coupling exists with another mode of vibration such that more energy is available to increase the magnitude of oscillation. In ground resonance, the blade lag motion is coupled with the in-plane motion of the hub and the rigid body motion of the fuselage in contact with the ground. Sufficient structural, landing gear and blade lag damping must be available to reduce the magnitude of the oscillations or the engines will provide not only the energy to turn the rotors but also the energy to increase the vibrations. Air resonance is a similar phenomenon that occurs when the helicopter is airborne. Both ground and air resonance are potentially destructive to the helicopter.

An example of the effect of negative damping on the system is rotor flapping stability in forward flight. As the advance ratio of the rotor increases, the system damping can, under certain conditions, decrease until it becomes negative. At this point the negative damping actually becomes a driving force in the rotor system with the airstream providing the energy source to increase the magnitude of oscillations until unbounded growth occurs.

### **C. ROTOR SYSTEM MODELING AND ANALYSIS PROCEDURES**

The first step in the rotor system analysis is to develop a model based on the equations of motion for a single rotor blade and the rotor hub. For this paper, two modeling methods are applied to an example rotor blade and hub assembly. The second step in the analysis process is to validate the model(s). The two methods chosen for this paper are compared to the exact solution for non-rotating beam vibrations and then contrasted, one to the other, for the case of rotating beam vibrations. In the third step, forces are applied to the models and they are again contrasted. One simple and effective modeling method is chosen and extended to the more complex case of a twisted, tapered and coupled rotor blade capable of various hub attachment schemes. In the fourth step, the single blade and hub case is expanded to a multiple blade system where the stability of the system is analyzed for the selected cases of flapping stability in forward flight and ground resonance.



## II. ROTOR BLADE STATIC AND DYNAMIC FREE RESPONSE

### A. ROTOR BLADE MODELING METHODS

#### 1. Exact Method

The process of developing a model for rotor blade vibrations begins with the observation that the blade physically resembles a long thin beam. If the rotor blade is assumed to be non-rotating, untwisted and uniform in mass and stiffness properties, the theory for simple beams may be applied to it. The Bernoulli-Euler beam theory relates applied forces to pure bending displacements and ignores effects of rotary inertia and shear deformation. The theory is valid as long as the blade is much longer than it is thick and the higher modes of vibration, where the rotary inertia and shear deformation effects are not negligible, are disregarded. Under these conditions, Bernoulli-Euler beam theory provides an exact solution for the continuous rotor blade.

Appendix A illustrates a brief derivation of the exact solution to the fourth-order differential equation of motion given by Bernoulli-Euler beam theory. This derivation provides equations for the displacement of the beam along its length and the natural frequencies of vibration. The displacement equation, **Equation A-5**, represents the mode shape of vibration at each natural frequency. The frequency and mode information determined from the exact method form the standard to which two other approximate methods of modeling will be compared.

Once the rotor blade begins to rotate, the exact solution described is no longer valid and no other closed form solution is available. The rotation of the blade results in a stiffening effect due to centrifugal force that is not constant along the blade. The centrifugal force term is governed by the relation:

$$C.F.(r,t) = \Omega^2 \int_r^L m \phi d\phi$$

**Equation 2- 1**

where the radius  $r$ , upon which the centrifugal force term is dependent, is itself dependent upon the mode shape of the beam. The inter-dependence of mode shape, frequency and centrifugal force is the reason why a closed form solution is not possible for the case of a rotating beam. To handle the rotating blade case, approximate methods must then be used.

## 2. Myklestad's Method

### a. Concept

The first approximate method used to model the rotor blade is Myklestad's method. This method is based on a lumped parameter discretization of a continuous system. The rotor blade model developed using this method is composed of a finite number of rigid masses connected by massless elastic elements of uniform stiffness. This lumped parameter method results in a replacement of the differential equations of motion by corresponding finite difference equations. The development of the modeling equations is accomplished in **Appendix B**. The equations relate each succeeding element from tip to root in terms of tip values, resulting in a transfer matrix  $[\alpha(\omega)]$  of the form:

$$\begin{bmatrix} Z \\ \theta^F \\ M^F \\ S^F \end{bmatrix}_{\text{root}} = [\alpha(\omega)]^{4 \times 4} \begin{bmatrix} Z \\ \theta^F \\ M^F \\ S^F \end{bmatrix}_{\text{tip}}$$

Equation 2- 2

where **Equation 2-2** represents only the flatwise motion of the rotor blade. The moment and shear are constrained to be zero at the tip. When the root boundary conditions described in **Appendix B** are applied, the result is two equations in terms of the two remaining tip unknowns, slope and displacement. The equations developed in **Appendix B** also include the edgewise motion and twist coupling terms which expand the transfer matrix to an eight by eight matrix. The solution process is identical to the flatwise only case. Torsional motion of the blade is not included in the coupled equations of **Appendix B** since the effects are usually small in contrast to flatwise and edgewise effects. Inclusion of torsional effects expands the transfer matrix to five by five in dimension.

To find the natural frequencies of the rotor blade, the applied forces are set to zero and a value for the excitation frequency  $\omega$  is assumed. The transfer matrix is formed and its determinant is calculated. If the determinant is not zero, a new value for the excitation frequency is assumed. This process continues until a zero is returned. This zero determinant state defines a natural frequency for the rotor blade for the stated boundary conditions. **Figure 2-1** depicts typical determinant residuals for assumed values of the excitation frequency of a non-rotating rigid rotor blade. The curve of residuals is continuous to infinity but has been clipped to view three axis crossings corresponding to the first three blade flatwise bending natural frequencies at approximately 5.5 rad/sec, 16.5 rad/sec and 33 rad/sec respectively. Each natural frequency

may be calculated without knowledge of any other natural frequency. This is not the case for many other methods.

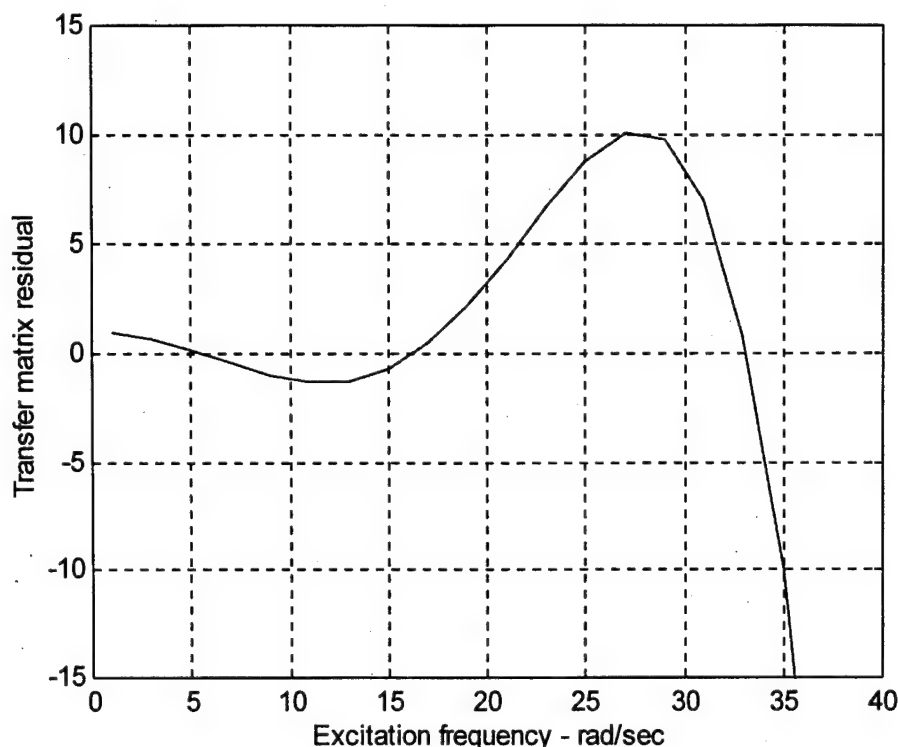
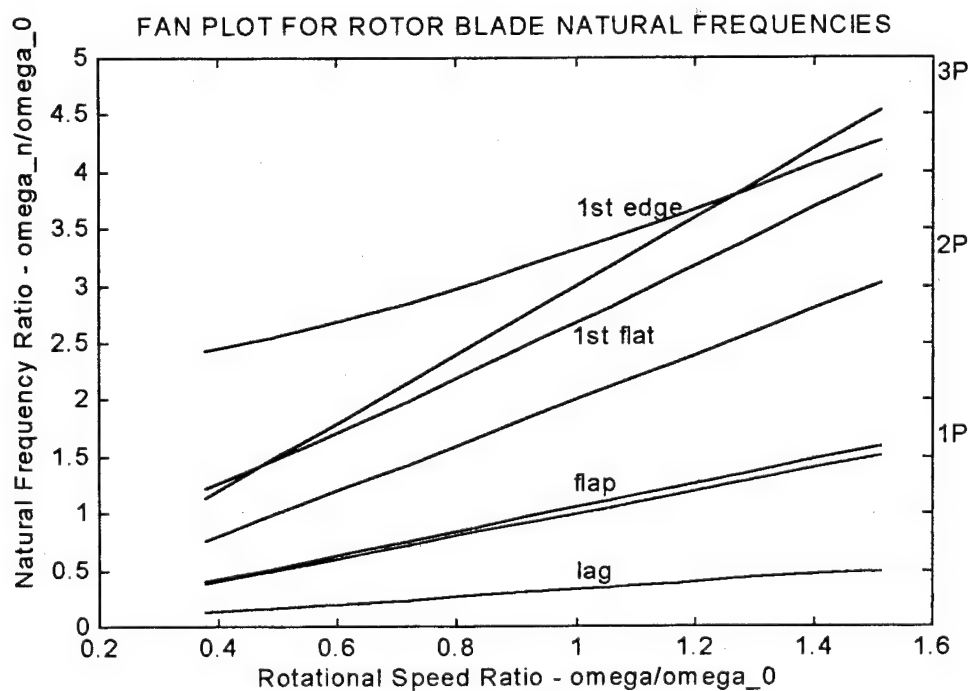


Figure 2- 1 Determinant residual for a typical non-rotating rigid rotor blade

#### *b. Software*

For the method developed in **Appendix B**, two software subroutines work in concert to determine the natural frequencies of a rotor blade. The subroutines are **mykbis.m** and **bldfan.m**, both developed by the author in MATLAB language and incorporated in the Dynamics section of the JANRAD software. Both subroutine texts are included in **Appendix F**. The subroutines are accessed through the Dynamics section of JANRAD by way of the **dynam.m** subroutine. The user inputs the basic helicopter configuration data into the main JANRAD program according to the directions described in Nicholson, 1993. The user then selects the Dynamics section from the menu and is prompted to input either a new file of blade data or the name of an old blade data file according to the directions given in Cuesta (1994). The blade data file consists of weight, chord and elastic stiffness values for each blade station along with the type of blade attachment (root boundary condition).

If the user elects to have a Southwell ,or fan, plot of natural frequencies generated for the input blade data, the **mykbis.m** subroutine requests the maximum excitation frequency desired and the value of the lag damping coefficient if a lag damper is to be included in the analysis. The maximum excitation frequency determines the number of modes that are calculated. **Mykbis.m** calls the **bldfan.m** subroutine for each excitation frequency step at a given rotor speed until a sign change in the returned residual is detected. The **bldfan.m** subroutine applies the Myklestad equations to the input blade data and returns a value for the determinant residual to the **mykbis.m** subroutine. A bisection routine then finds the values of excitation frequency that correspond to a determinant residual value of zero. This process continues until the maximum excitation frequency is reached. **Mykbis.m** then increments the rotor speed to the next value and repeats the process. The **mykbis.m** subroutine determines the natural frequencies for all modes up to the maximum excitation frequency at rotor speeds ranging from 10 percent to one hundred fifty percent of normal operating speed. The natural frequency versus rotor speed plot is non-dimensionalized by the normal operating speed of the rotor and output to the screen for display. Included on the graph are lines depicting n/rev excitation frequency ratios. **Figure 2-2** is an example of the **mykbis.m** program output showing a fan plot of the first four natural frequency ratios for a typical rotor blade along with the first three n/rev excitation frequency ratios.



**Figure 2- 2 Fan plot for a typical rigid rotor blade**

From the design point of view it is important to remove the rotor blade natural frequencies from the n/rev excitation frequencies over the normal rotor operating speed range. This ensures that the blades will be free of resonance under normal operating conditions. Rotor blade natural frequencies are adjusted by changing mass and/or stiffness characteristics at strategic points in the blade. **Figure 2-2** shows that the example rotor blade remains clear of excitation frequencies over a wide range of rotational speed ratios for the case of first flatwise bending and first chordwise bending modes. There is only a small amount of separation between the rigid flap mode and the 1/rev excitation frequency, but this mode is not normally a problem since it is heavily damped.

### 3. Finite Element Method

#### a. Concept

The second approximate method used to model the rotor blade is the finite element method. Many different techniques are associated with the term "finite element method". In this paper the term denotes a consistent approach where the mass, damping and force matrices are consistent with the stiffness matrix. This means the mass, damping and force characteristics, like the stiffness characteristics, are spread throughout the element instead of being assigned a discrete position on the element. The result is a rotor blade that has been subdivided into a number of continuous elements. The method is approximate because it defines the displacement at any point in an element by interpolation between a finite number of nodal displacements. The derivation of these interpolation functions and the element equations of motion are included in **Appendix C**. Although there is no requirement that blade properties be constant within an element, the derivation in **Appendix C** assumes this to be true.

Solution of the finite element eigenvalue problem is accomplished using the power method with matrix deflation. The power method assumes that the solution to the eigenvalue problem is given by:

$$([D] - \lambda[I])\{u\} = \{0\} \quad \text{where} \quad [D] = [K]^{-1}[M]$$

**Equation 2- 3**

The equation consists of linearly independent eigenvectors of the dynamical matrix  $[D]$ . An initial trial vector is chosen for the eigenvector, or mode shape,  $\{u\}$  and multiplied by the dynamical matrix such that:

$$[D]\{u\} = \lambda\{u\}$$

**Equation 2- 4**

The resulting vector  $\lambda\{u\}$  is normalized by its largest value and then multiplied by the dynamical matrix again. This iteration process continues until the  $\lambda\{u\}$  vector converges to a consistent value. The eigenvalue  $\lambda$  is then equal to the inverse of the largest value of the  $\lambda\{u\}$  vector and the mode shape is equal to the normalized  $\lambda\{u\}$  vector. The eigenvalue is related to the natural frequency according to:

$$\lambda_i = \frac{1}{\omega_i^2} \quad i = 1, 2, \dots, n$$

#### Equation 2- 5

The resulting natural frequency and mode shape are always for the dominant mode of the system. In order to determine the next higher mode, all trace of the current dominant mode must be removed from the dynamical matrix. The process of removing this mode is called "matrix deflation". To deflate the

$$\{u\}_i^T [M] \{u\}_i = 1$$

#### Equation 2- 6

dynamical matrix, the eigenvector is first normalized such that:

and the new dynamical matrix is calculated from the old one according to:

The new dynamical matrix has the same eigenvalues as the old one except that the eigenvalue for the old dominant mode is now equal to zero. Applying the power method to the new dynamical matrix reveals the next dominant mode of the system. The process of applying the power method and deflating the dynamical

$$[D]_{i+1} = [D]_i - \lambda_i \{u\}_i \{u\}_i^T [M]$$

#### Equation 2- 7

matrix can be continued as far as required, but each mode must be determined in order. In other words, the third mode cannot be determined before the first and second modes have been calculated and removed from the dynamical matrix. (Meirovitch, 1986)

### ***b. Software***

A stand alone function called **rotor.m**, developed by the author in MATLAB code, performs the steps required to model an untwisted, uncoupled rotor blade with uniform mass and stiffness properties by the finite element method. This function is included as **Appendix F**. **Rotor.m** requires inputs for the mass/length, the bending stiffness  $EI$ , the radius of the blade, the number of elements desired, the hinge offset (flapping or lag depending on the blade incidence angle input), the normal rotor speed



desired, the blade incidence angle (zero degrees for pure flap motion and ninety degrees for pure lag motion), the number of modes desired and the boundary conditions to be applied at the root. A sample input for the **rotor.m** function is:

`rotor(.3,200000,25,20,0,30,0,3,1)`

where the mass/length is 0.3 slug/ft, the bending stiffness is 200000 lb-ft<sup>2</sup>, the rotor radius is 25 ft, the model consists of 20 elements, there is no hinge offset, the normal rotor speed desired is 30 rad/sec, the angle of incidence of the blade is zero degrees (corresponding to pure flapping motion), the first three modes are to be included and cantilever root boundary conditions are to be applied to the blade (a zero input is required for articulated rotor blade root boundary conditions).

The function begins by developing the element mass and stiffness matrices with the given mass and stiffness properties assumed to be constant through the element. These matrices are assembled into the corresponding global matrices comprised of the desired number of elements. The element centrifugal stiffness matrix is then calculated for each element in turn, based upon the current rotor speed and the elements distance from the center of rotation, and assembled into a global matrix. A total stiffness matrix is formed from the addition of the elastic and centrifugal stiffness matrices. The dynamical matrix is determined according to **Equation 2-3** and contains entries for the degrees of freedom, representing slope and deflection at the nodes, equal to twice the number of elements for the cantilever root case and twice the number of elements plus one for the articulated root case. The extra degree of freedom in the articulated root case is required to allow rotation of the blade around the hinge. All the elements are now in place to solve the eigenvalue problem and determine the rotor blade natural frequencies.

The **rotor.m** function assumes an initial mode shape component of one for each degree of freedom in the model. The power method is applied to the dynamical matrix utilizing the initial mode shape and refining the resulting vector until it converges to the accuracy of the computational system. The natural frequency of the current mode is determined according to:

$$\omega_{ni} = \sqrt{\frac{1}{\lambda_i} - \Omega^2 \sin^2 \theta}$$

**Equation 2- 8**

where the extra term  $\Omega^2 \sin^2 \theta$  is introduced, without derivation, as the reduction in the natural frequency due to an increase in the incidence angle of the blade (Hoa, 1979). The mode shape is then normalized and the dynamical matrix deflated. This process continues until the desired number of modes have been determined. The function then increments to the next rotor speed and repeats the mode calculations. Once the maximum rotor speed is reached, the function outputs the natural frequencies to the display as a fan plot of frequency ratios normalized to the normal rotor operating speed as in the **mykbis.m** program.

Figure 2-3 is a typical output from the rotor.m function for the rotor parameters listed in the example input line above. In Figure 2-3, the natural frequency of the second flatwise bending mode passes close to the 5/rev excitation frequency ratio at the normal rotor operating speed. This natural frequency would need to be adjusted based upon the significance of the rotor response in that particular mode to that particular excitation frequency.

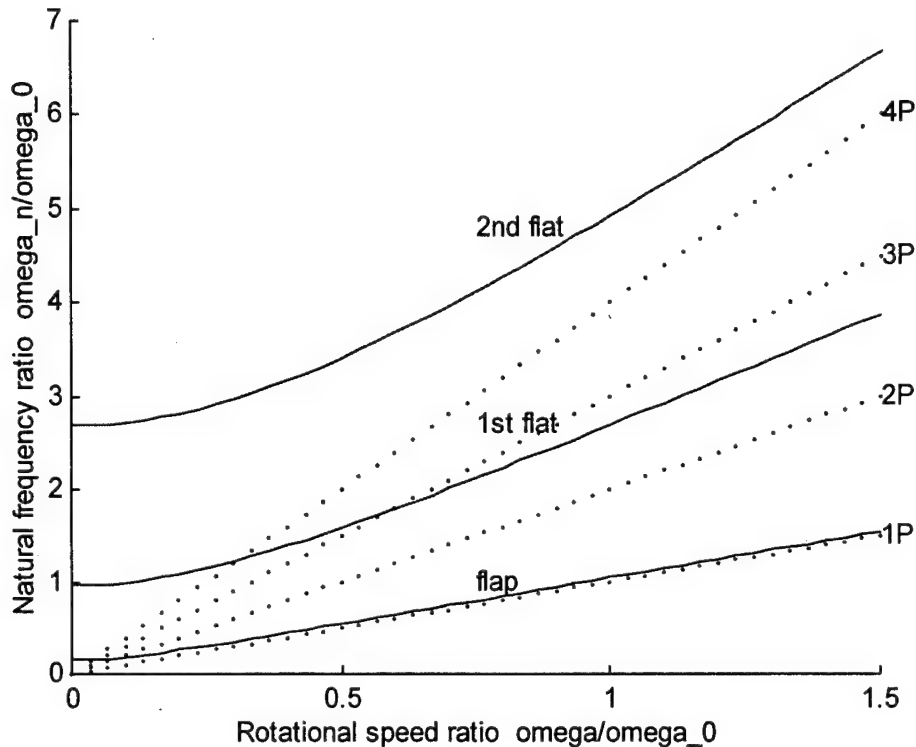


Figure 2- 3 Fan plot output for the rotor.m function sample input

## B. ROTOR BLADE MODEL COMPARISON

### 1. Non-rotating Uniform Blade Case

The rotor blade models by finite element method and Myklestad's method can only be compared to an exact method for the case of a non-rotating uniform blade. The purpose of such a comparison is to illustrate the effectiveness of the modeling method in approximating the simplest case of vibration for a rotor blade. The natural frequencies and corresponding mode shapes are the only information upon which a comparison can be made. The rotor blade to be modeled has a cantilever (rigid) root condition and the

following characteristics: mass/length = 0.3 slugs/ft, rotor radius = 25 ft, hinge offset = 0 ft, elastic bending stiffness coefficient  $EI = 200000 \text{ lb-ft}^2$ , rotational speed = 0 rad/sec.

Each method is compared by using 15 element and 50 element models to determine the natural frequencies of the first 15 modes and the mode shapes of the first five modes. Table 2-1 lists the natural frequencies for the first 15 modes as determined by the exact method and by 15 element and 50 element Myklestad and finite element models.

mode	exact	finite-15	finite-50	Myklestad-15	Myklestad-50
1	4.59330311	4.5933039	4.59330312	4.5978	4.59374
2	28.78573924	28.78593	28.785741	28.877	28.7949
3	80.60090339	80.605	80.600937	81.07	80.642
4	157.9456017	157.976	157.94586	159.7	158.06
5	261.0953968	261.23	261.0966	267.0	261.35
6	390.0313124	390.48	390.0352	407.6	390.54
7	544.7544818	545.96	544.765	592.0	545.70
8	725.2648424	728.0	725.289	no solution	726.96
9	931.5623975	937.3	931.61	no solution	934.51
10	1163.64715	1174.5	1163.75	no solution	1168.6
11	1421.51909	1440.5	1421.70	no solution	1429.2
12	1705.17823	1736.3	1705.50	no solution	1722.0
13	2014.62456	2062.1	2015.1	no solution	no solution
14	2349.85809	2414.2	2350.7	no solution	no solution
15	2710.87881	2761.1	2712.1	no solution	no solution

Table 2- 1 Natural frequencies for example rotor using 15 and 50 element models

In Table 2-1, the number of significant digits included is based on the relative accuracy of the approximate frequency as compared to the exact frequency. To keep the comparison equitable, double precision computational schemes are not used in the determination of the approximate frequencies. Neither approximate method gains a distinct advantage from the use of higher precision.

One point of emphasis is the "no solution" entries in the table. These entries indicate the Myklestad method is unable to resolve the frequency at this point. In the case of the 15 element Myklestad model, the simple cantilever beam deflection characteristics assumed by the Myklestad elastic element described in Appendix B require at least two elements per mode desired. Since only 15 elements are used, only the first seven modes can be determined. The 50 element Myklestad model is not limited by the number of elements in this comparison. As the frequency increases with each mode, the determinant residual values rapidly increase and the slope of the residual at the zero point approaches 90 degrees. At some point, without the use of higher calculation precision, the Myklestad 50 element model becomes unable to resolve the natural frequency. This resolution breakdown in the `mykbis.m` program

usually occurs, regardless of rotor blade physical properties tested, in the vicinity of 2000 rad/sec frequency for the cantilever root condition. This resolution limitation turns out not to be as important as it would appear, as will be discussed later in the paper.

The percent absolute error of each frequency determined by approximate methods as compared to the actual frequency is depicted in Figure 2-4.

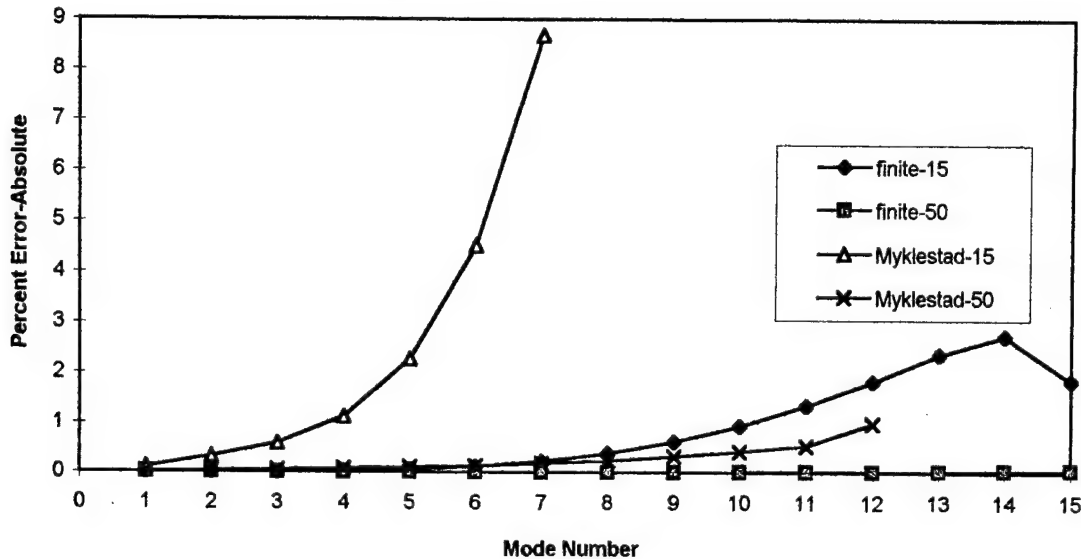


Figure 2- 4 Percent error in natural frequency calculation for a cantilever beam.

Figure 2-4 shows the finite element method is able to approximate the exact frequency more closely, element for element, than the Myklestad method. The finite element method does not suffer from the requirement to have at least two elements for each mode desired, although there should be at least one element for each mode desired. The finite element method is also able to resolve natural frequencies without limit. This resolution capability must be tempered by the fact that each mode must be calculated in order from first to last. As each succeeding mode is determined and the dynamical matrix is deflated, roundoff error begins to build and reduce the accuracy of the estimation. This loss of accuracy is apparent in Figure 2-4 for both the 15 and 50 element finite element models. Another way to visualize the accuracy of the approximate methods is to compare the mode shapes for corresponding natural frequencies. Figure 2-5 shows a comparison of the third mode shape as determined by each method. The finite element method determined a mode shape for the third mode that is indistinguishable from the exact mode shape. The Myklestad method determined mode shape has separated from the exact shape, most noticeably at the anti-nodes. The natural frequency of vibration calculated by an approximate method cannot be the same as the exact frequency if the mode shape is not also identical to the exact mode shape.

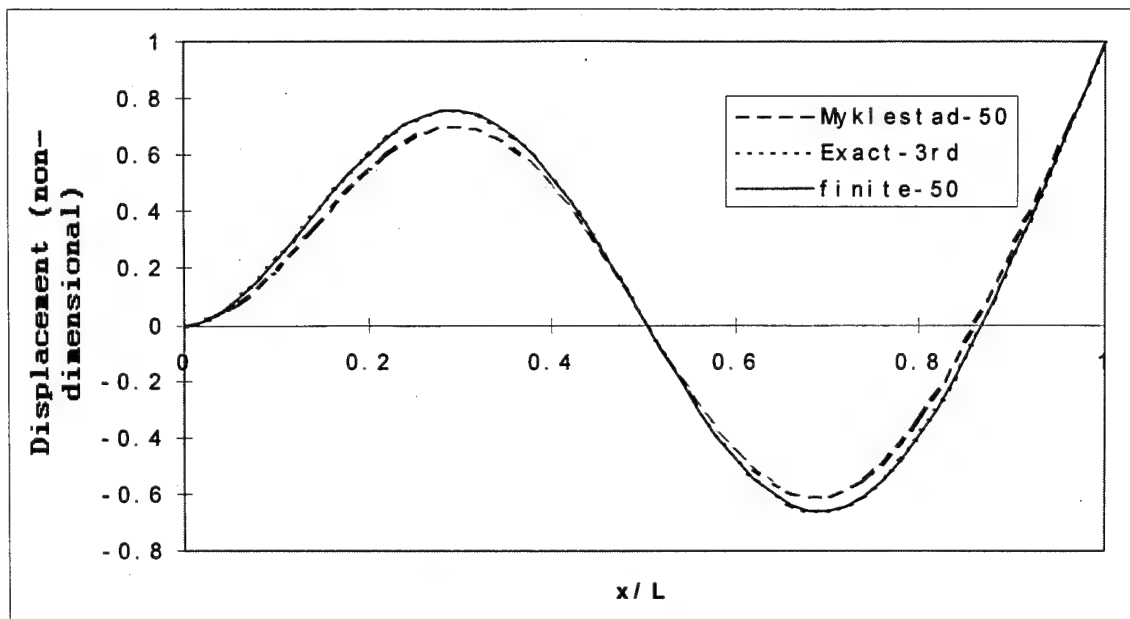


Figure 2- 5 Third mode shape for a cantilever beam.

In modeling a non-rotating uniform blade, the finite element method is at an advantage for accuracy. The Myklestad method requires two to three times as many elements to resolve the natural frequency to the same accuracy as the finite element method and is unable to resolve higher modes without an increase in the precision of the computational scheme. Neither method is more difficult to implement than the other for this simple case.

## 2. Rotating Uniform Blade Case

As stated previously, the exact solution is no longer valid and no other closed form solution is available once the rotor blade begins to rotate. The centrifugal force becomes a significant factor, increasing the natural frequencies of each mode as the rotational speed increases. The comparison of the Myklestad method to the finite element method now becomes more qualitative in nature. If the same uniform blade assumptions and cantilever root conditions are kept, the methods can be contrasted by how many elements are required to approach a stable frequency value.

At this point, a discussion of the trend for solutions by each method is necessary. The finite element method, in general, over estimates the natural frequency for any given mode in the non-rotating blade case. As elements are added to the model the approximation converges to the exact value from above. If this trait is assumed to carry over to the rotating blade case, the finite element method should

always provide a maximum value possible for the exact natural frequency. In the Myklestad method, the lumping of masses is largely an arbitrary process. Changes in the distribution of mass throughout the blade have a significant effect on the natural frequency estimation. This effect becomes more pronounced with increasing mode number and the inclusion of centrifugal force. In fact, the value of the lumped mass representing the end of the rotor blade is determined from either a full element length or half element length, depending upon which configuration provides better results. Therefore, a similar general statement cannot be applied to the Myklestad method. In this paper, if the Myklestad approximation for the natural frequency is higher than the finite element approximation, the Myklestad approximation is assumed to have a larger error. If the Myklestad approximation is lower than the finite element approximation, no conclusion is drawn.

Both approximate methods are applied to a uniform, rotating blade at three rotational speeds of 10, 20, and 30 rad/sec using 15, 30 and 50 elements consecutively at each speed. The approximate frequencies are determined for the first ten modes under the given speed and element conditions. Comparison of the frequency convergence behavior for both approximate methods gives insight to the effectiveness of the models. **Figures 2-6 thru 2-8** show the finite element frequency approximation for the second, sixth and tenth modes at a rotational speed of 10 rad/sec decreasing toward a stable value from above, as previously assumed. The percentage of change in the frequency approximation from 15 elements to 30 elements and from 30 elements to 50 elements is only 0.0006% and 0.0003% respectively for the second mode, 0.102% and 0.006% respectively for the sixth mode and 0.85% and 0.056% respectively for the tenth mode. As seen in the non-rotating blade case, the accuracy of the frequencies determined by the finite element method is decreasing with increasing mode number, but this is offset by the rapid convergence achieved through increasing the number of elements in the model. The finite element model convergence appears relatively insensitive to increasing rotational speed as illustrated by the change in frequency approximation from 15 elements to 30 elements for the tenth mode where 0.85%, 0.80% and 0.74% are the percentage change for 10, 20 and 30 rad/sec respectively.

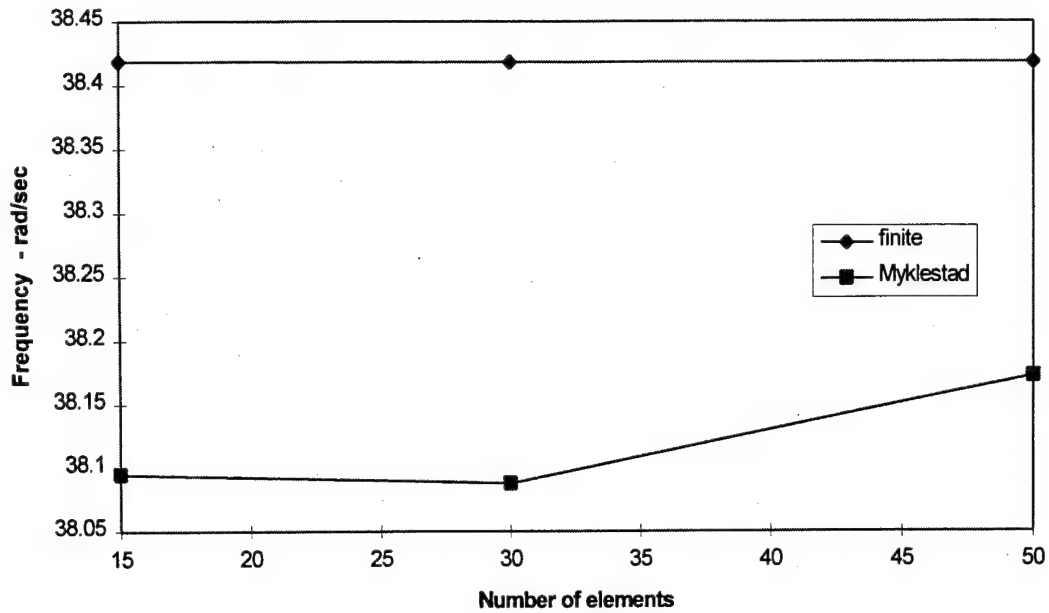


Figure 2- 6 Cantilever beam second mode frequency approximation.

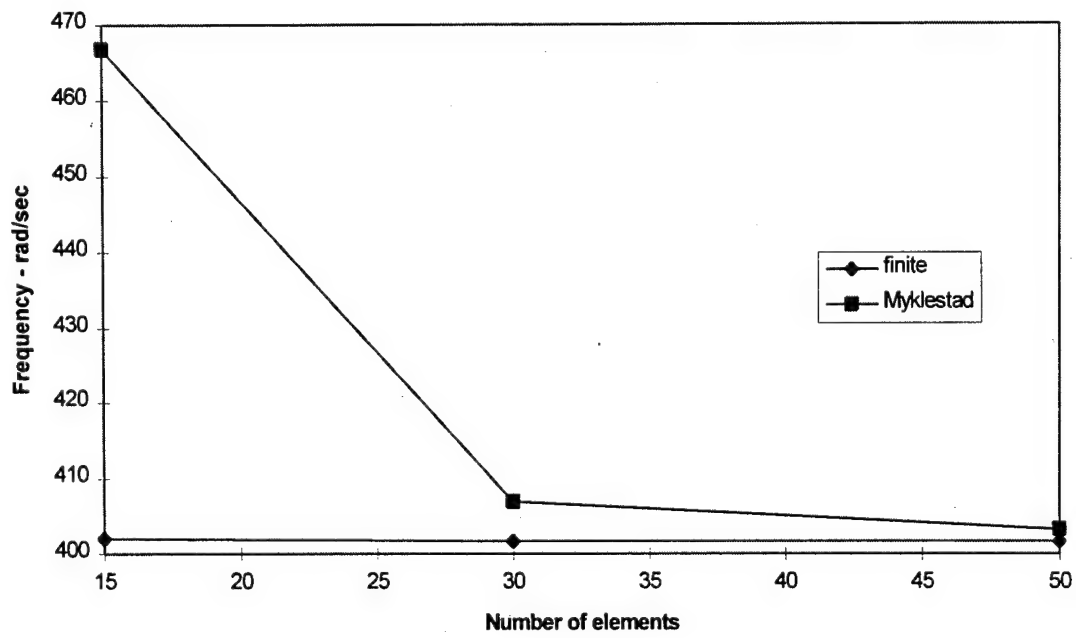
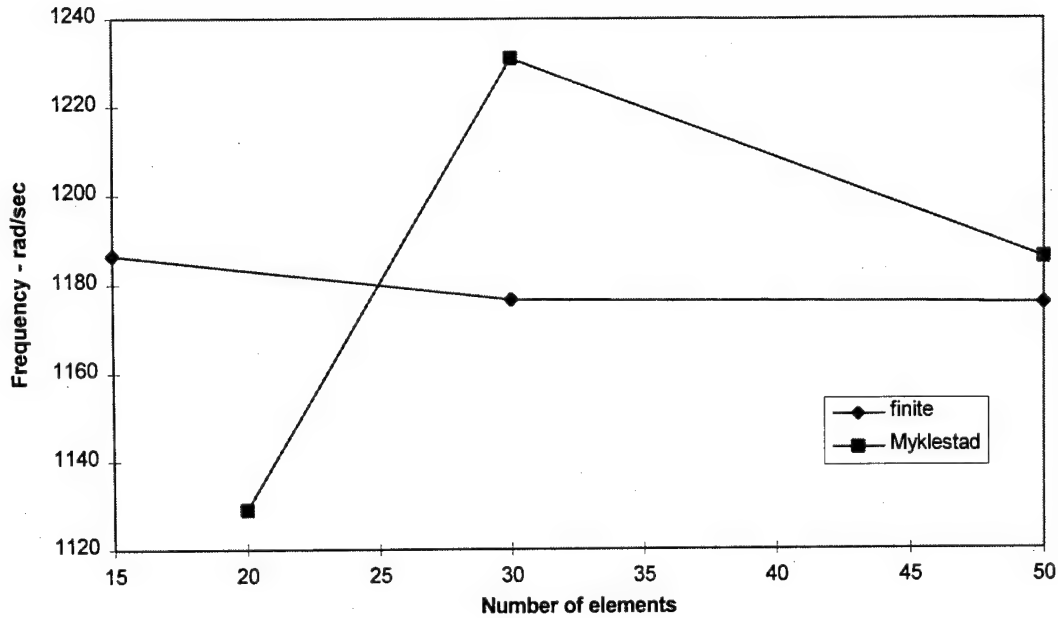


Figure 2- 7 Cantilever beam sixth mode frequency approximation



**Figure 2- 8 Cantilever beam tenth mode frequency approximation.**

In contrast to the finite element model approximations, the behavior of the Myklestad model approximations is not so clearly defined. In **Figure 2-6**, the frequency approximations appear to be approaching a stable value for the second mode from below and remain below the finite element approximations as the number of elements in the model increases. In **Figure 2-7**, the Myklestad frequency approximations for the sixth mode decrease by 14.7% when the number of elements is increased from 15 to 30 and an additional 0.94% when the number of elements is increased to 50. The Myklestad approximations remain above the corresponding finite element values. **Figure 2-8** shows the Myklestad frequency approximations crossing from beneath the finite element values to above and then decreasing to the final value. The first data point in this figure utilizes 20 elements instead of 15 because a minimum of 20 elements are required to produce a valid frequency approximation for the tenth mode as previously discussed.

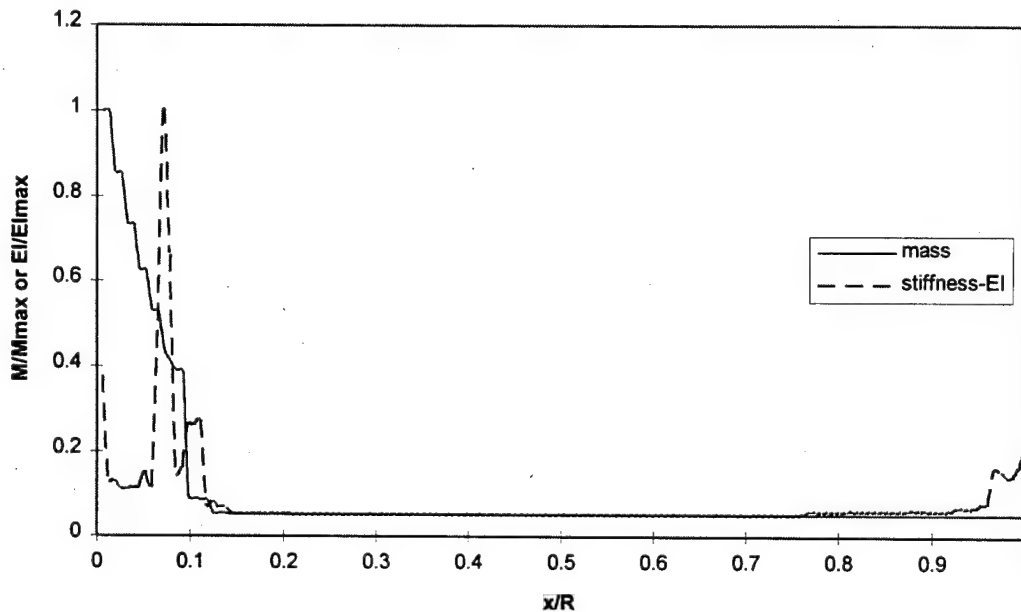
The Myklestad approximations show little sensitivity to the increase in rotational speed as was true with the finite element model. In all cases, the final value for the frequency determined by the Myklestad model is within 1.7% of that determined by the finite element model for 50 elements. The rotating blade comparison of the two methods shows, as in the case of the non-rotating blade, that up to three times as many elements are required by the Myklestad model in order to provide an approximation for the natural frequency that is within about 1.0% of a stable value. The comparison also highlights the



predictable nature of the finite element model's behavior as opposed to the more random nature of the Myklestad model.

### 3. The Effects of Non-uniformity

The previous two test cases used for comparison of the approximate methods are based upon the assumption that the rotor blade is of uniform properties and is not twisted or tapered. This assumption does not reflect the true nature of rotor blades. **Figure 2-9** illustrates the highly non-uniform nature of an actual rotor blade. Both the mass and stiffness parameters change dramatically in the vicinity of the rotor blade root. The reason for these abrupt changes is a structural requirement to attach the blade securely to the hub and to provide for the dynamic balance, in-plane damping and control of the blade, while still allowing the blade freedom to move within the constraints of the blade root attachment scheme.



**Figure 2- 9** Non-dimensional mass and stiffness distributions.

From **Appendix V**, the mass and stiffness terms are assumed constant throughout the element for the consistent finite element method derived. If this is not the case, the mass and stiffness terms become dependent upon radial position. This means the mass and stiffness terms in **Equations C-10** and **11** must be integrated over the length of the element. In order to perform the integration, the mass and stiffness must be continuous over the element such that the terms may be represented by a polynomial of some order. Representing the mass and stiffness of the blade in **Figure 2-9** by a polynomial or series of local polynomials for each element reduces the accuracy of the estimation while drastically increasing the

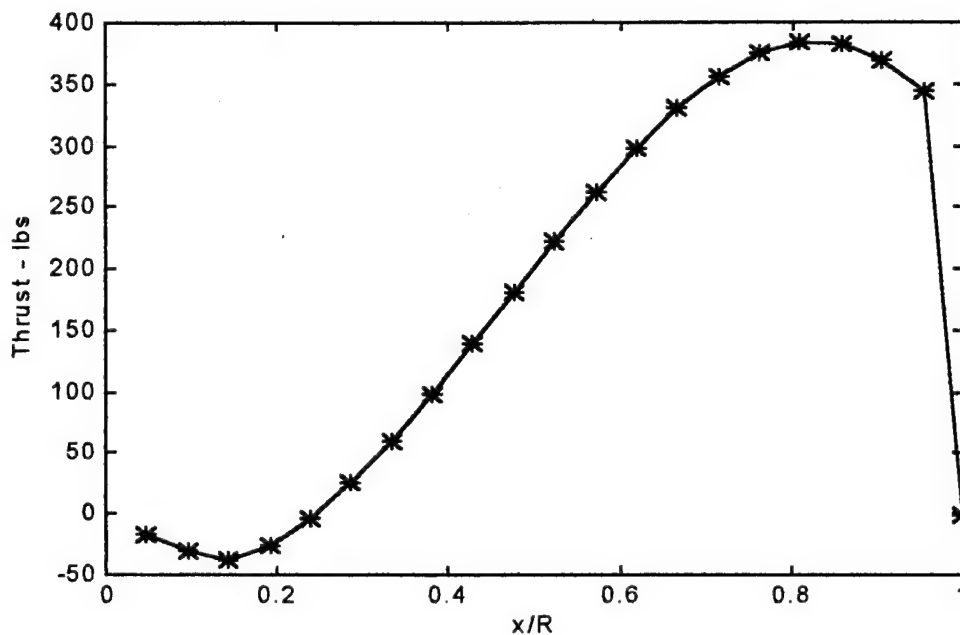
complexity of the modeling equations. An alternative to the increased complexity associated with polynomial distributions is to use an average value over the element. This can be used effectively, but some of the accuracy that makes the finite element method so attractive is now lost.

In modeling a non-uniform rotor blade, the Myklestad method developed in **Appendix B** provides a simple model based upon lumped parameters. These lumped parameters are capable of handling discontinuities in the properties with greater ease than the finite element method developed. In fact, the model complexity does not change despite the most dramatic changes in rotor physical parameters. The accuracy of the approximation is merely a function of the number of elements used in the model.

### III. ROTOR BLADE DYNAMIC FORCED RESPONSE

#### A. FORCED RESPONSE OF THE ROTOR BLADE

Having established methods to study the free response of the rotor blade, the next logical step is to apply forces to the blade and develop methods to determine its forced response. The forced response of the rotor blade is based upon steady state conditions where the thrust and drag forces on the blade are replaced by an equivalent series of harmonic excitation forces. These excitation forces are determined from the harmonic decomposition of the rotor blade thrust and drag calculated by the rotor performance analysis section of the JANRAD program. The thrust and drag forces calculated by the rotor performance analysis subprogram are output as two matrices with the entries relating thrust or drag to radial position on the blade and blade azimuth. **Figure 3-1** shows a typical thrust distribution calculated by the rotor performance analysis section for a rotor blade at zero degrees azimuth.



**Figure 3- 1** Typical thrust distribution for a rotor blade at zero degrees azimuth.

The stars on the thrust plot indicate the thrust values calculated by the performance subprogram at discrete radial positions based upon the number of elements desired for the blade model. **Figure 1-1** shows how the thrust changes at one blade radial position according to azimuth.

Once the thrust and drag forces are harmonically decomposed, these harmonic force components are individually applied to the rotor blade model at the centers of the blade elements. One fundamental rule for the forced response is that the blade will respond at the same frequency as the excitation force but with some dynamic magnification factor applied to the amplitude and some change in the phase angle of the response. The blade's response to each harmonic excitation component must then be combined to determine the total response of the blade subjected to steady thrust loading. This total response takes the form of the displacement of the blade as a function of radial position and azimuth for the applied flight loads.

In the process of designing a rotor blade or evaluating a preliminary design's dynamic characteristics, information on the stress resultants internal to the blade is as important as the magnitude of the blade response. Designers typically want to know the distributions of shear force and moments along the blade to ensure critical stress levels are not exceeded under dynamic loading. A forced response model for a rotor blade should provide not only information on the displacement of the blade but the shear and moment distributions as well. In the next two sections, the finite element method and Myklestad method are subjected to harmonic loading to evaluate each model's effectiveness.

## **B. FORCED RESPONSE USING THE FINITE ELEMENT METHOD**

As indicated in the free vibration case, the basic finite element method provides a very accurate natural frequency model under non-rotating conditions with rapid convergence to a stable frequency value under rotating conditions provided the rotor blade is assumed to be uniform. However, with non-uniform blade properties, the method requires that the non-uniformity be representable by a polynomial distribution, i.e. no discontinuities are allowed. The use of polynomials to represent physical properties results in increased complexity of the element equations of motion. Forces applied to the finite element model are not subject to these same constraints because the method accommodates distributed or point forces and moments as long as they can be resolved to the nodal points.

A uniformly distributed excitation force is applied to the finite element model using a consistent approach as described in **Appendix C**. For some other loading condition, the applied forces and moments are resolved to the nodes and incorporated in the forcing vector. Before forces are applied to the model, the natural frequencies are determined by the power method using matrix deflation. The mass and stiffness matrices are then normalized and the equations of motion are transformed from a nodal coordinate system to a modal coordinate system by a transformation matrix composed of orthogonal eigenvectors (modes) corresponding to the natural frequencies of the blade.

Under the assumption that the blade responds to harmonic forcing at the same frequency as the excitation, **Equation C-26** calculates the response for each individual mode and superposes them in a single step to return the total response of the blade to that particular harmonic force. This modal response is then transformed back into the nodal coordinate system using the inverse of the transformation matrix. Once the nodal response is determined for a particular excitation, the interpolating functions are applied to this nodal response through **Equation C-6** to determine the displacement of the blade over each element.

The moment distribution of the blade is determined by taking the second derivative of the displacement equation (**Equation C-3** or alternatively **C-6, Appendix C**) and relating it to the moment using the simple beam theory where:

$$M(x) = EI \frac{d^2 w(x)}{dx^2}$$

**Equation 3- 1**

Taking the second derivative of displacement equation reveals that the moment distribution is a linear function over the element. This is an important point because the magnitude of the moment at each node is preserved from element to element, but the slope of the moment distribution from one element to the next is discontinuous. Similar conditions exist when attempting to calculate the shear force distribution. The shear force in an element is given by:

$$S(x) = EI \frac{d^3 w(x)}{dx^3} - T(x) \frac{dw(x)}{dx}$$

**Equation 3- 2**

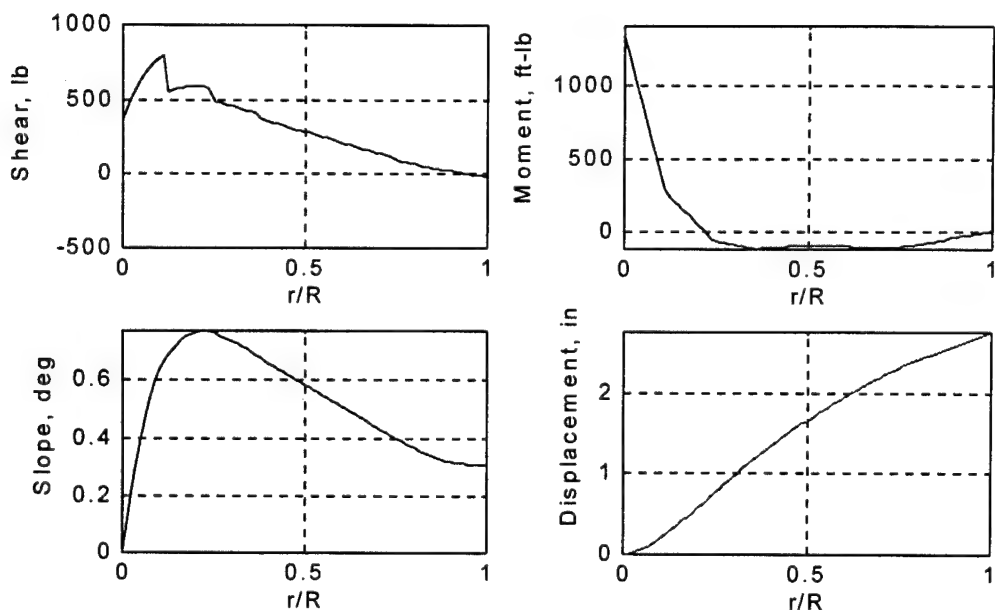
where the third derivative of the displacement equation is a constant, the tension is decreasing over all the elements as the radius squared and the first derivative of the displacement equation is also a squared term inside the element. The result is a shear distribution that is also discontinuous from one element to the next. In fact, the value for the shear force that is most representative over the element is found at the element midpoint.

**Rtrtot.m**, a MATLAB function developed by the author to illustrate the breakdown of the standard finite element method in determining the shear and moment distributions is included in **Appendix F**. **Rtrtot.m** uses the basic **rotor.m** function code to determine the natural frequencies of the blade prior to developing the damping and forcing matrices. The function transforms the blade equations of motion into modal coordinates, solves for the forced modal response of the blade one mode at a time and then

transforms the resulting modal displacement vector back into the nodal coordinate system. The total nodal response returned is the superposition of each of the individual nodal responses. The function then applies the displacement equation and its derivatives to the nodal response as appropriate to develop the displacement, slope, moment and shear distributions. The function produces these distributions as four individual plots. **Figure 3-2** is an example of the `rtrtot.m` function output for the following input:

`rtrtot(3,200000,25,8,0,20,0,5,1,20)`

where the entries are for a uniform mass of 0.3 slugs/ft, uniform stiffness of 200000 lb-ft<sup>2</sup>, blade radius of 25 ft, 8 elements, 0 ft hinge offset, rotational speed of 20 rad/sec, 0 degrees incidence angle, the first 5 blade natural modes to be included in the response, the blade has a cantilever root condition and the forcing frequency is 20 rad/sec.



**Figure 3- 2 Example rotor blade shear, moment, slope and displacement distributions.**

The distributions shown in **Figure 3-2** are for a uniform distributed force equal to 40 lb/ft applied over the length of the blade. As the finite element method applies higher derivatives of the basic displacement equation to determine the shear and moment, the plot of the resulting distribution becomes more disjointed. The moment distribution is a series of linear sections representing the moment distribution in an element. These sections are connected at the nodes but not necessarily matching in slope at the connection point. Discontinuities in the shear distribution are most obvious toward the blade root in **Figure 3-2**, where the trend of the shear distribution appears to pass through the midpoint of the element as

previously indicated. Although a better average value for the shear and moment distributions is attained by adding elements to the model, the discontinuities cannot be removed.

The finite element method of the basic form described in **Appendix C** does not provide an effective method for dealing with non-uniform rotor blades nor does it provide representative shear and moment distributions along the blade. The method is not very powerful in its basic form because it is based on the minimum order interpolation functions required to meet slope and displacement requirements at the nodes. To increase the capability of the method, the basic formulation is expanded to include higher order interpolation functions, a suitable set of which is described in **Appendix C**, and allowed to include discontinuities in the physical properties. Several authors have produced papers on the topic of increasing model flexibility and performance through variable-order finite element methods, including Hodges (1979) and Rutkowski (1980). Notwithstanding, the increased complexity of these methods precludes their easy formulation and application.

### **C. FORCED RESPONSE USING THE MYKLESTAD METHOD**

The free vibration case showed that the Myklestad based model, while generally less accurate than the finite element model for uniform conditions, is more flexible in its ability to handle discontinuities and non-uniformity in physical parameters. This statement holds true for the forced vibration case as well, but for one exception. The exception is a requirement that all excitation forces applied to the Myklestad model be steady or harmonic in nature. In other words, this method is not capable of handling random or impulse type excitations. This limitation does not normally present a problem while studying the steady state forced response.

**Appendix B** outlines the procedures for applying harmonic forces to the Myklestad model. In the equations developed in the appendix, the harmonically decomposed forces are applied in the complex form described in the Introduction section. The complex form is used to keep cosine and sine forcing terms separate throughout the calculation of the blade response. This results in a complex form for the blade response to each harmonic force applied. After all the forcing harmonics are applied and the response to each one is calculated, the total response for the blade is determined by superposing the individual harmonic responses on one another.

The only differences between the free vibration solution and the forced vibration solution of the Myklestad equations are that the applied forces are no longer zero, the excitation frequencies now correspond to the frequency of the harmonic excitation force applied and the determinant residual is no longer required to be zero. In the forced vibration solution process, the harmonically decomposed thrust and drag forces are applied in the complex form at the corresponding  $n/\text{rev}$  excitation frequency. The

matrix resulting from passing through the equations element by element from tip to root is used to solve for the unknown slope and displacement values at the tip. These values are re-entered into the matrix to determine the actual response of the blade at each station for each excitation force. The total response is determined by recombining all of the harmonics. This process results in distributions for the displacement, slope, moment and shear that vary with blade azimuth.

## D. JANRAD FORCED RESPONSE MODEL

The JANRAD program utilizes the Myklestad method to model the rotor blade primarily because of the method's flexibility, intuitive nature and direct calculation of displacement, slope, shear and moment distributions. The resolution of the model is not a key issue since more elements can always be added to refine the solution, provided the qualitative behavior of the blade is accurately reproduced. The subprogram **bldrev.m**, revised from Cuesta (1994) by the author, resides in the Dynamics section of JANRAD and implements the Myklestad rotor blade equations developed in **Appendix B**. **Bldrev.m** is included in **Appendix F**.

The Myklestad equations utilized by the **bldrev.m** subprogram include the effects of blade twist and in-plane forces due to flapping motion (in-plane Coriolis force) on the flap and lag motion of the blade. These effects are the primary means by which the flap and lag motions of the blade are coupled. The out-of-plane Coriolis force (out-of-plane force due to lag motion) and the torsional degrees of freedom have been neglected because the terms are generally small in magnitude. For preliminary design, these neglected terms do not have a large influence on the forced response solution. In fact, if the rotor blade is designed such that the center of lift and elastic center coincide, the torsional effects can be considered negligible.

### 1. Rotor Blade Model Input Requirements

The **bldrev.m** subprogram requires inputs from two sources. The first source is the main JANRAD performance section where the basic helicopter parameters are input. From this source, **bldrev.m** receives user input values for rotor radius, rotor speed, hinge offset, the number of blade model elements desired, rotor blade total twist and the number of blade azimuth positions desired. The performance section also provides **bldrev.m** with calculated values for the thrust force and drag force on the blade, blade steady coning angle and the collective pitch angle for the blade as measured at the standard position of 0.7 times the blade radius. The forces, determined from the helicopter physical parameters and flight conditions, are in a matrix form of thrust or drag force by radial position and azimuth position.



The second source of inputs for **bldrev.m** is the dynamics section of JANRAD. The **dynam.m** subprogram prompts the user to input an existing rotor blade parameter file name for editing or develop a new parameter file. The rotor blade parameter file contains information on the rotor blade's root attachment condition, weight, chord and elastic stiffness. The root attachment condition is an input value that indicates whether the blade root is considered to be articulated or rigid. The weight information is entered as a vector of weight per unit length values for each element. The chord information is for average blade chord for each element. The stiffness information is input as elastic stiffness  $EI$  or as Young's modulus  $E$  and first area moment of inertia  $I$  for each element and for both flatwise and edgewise (or chordwise) bending. An important assumption imposed is that the blade parameters are uniform over the length of the element. Discontinuities and highly non-uniform parameters are handled by adjusting the value of the parameter in a specific element or by increasing the number of elements used. The subprogram also prompts the user to input a value for the lag damper coefficient of damping. If no lag damper is incorporated, a zero entry is used. The lag damper is only available for the case of an articulated blade root condition. Once the rotor blade parameters are input, **dynam.m** saves the information in a data file for use in calculations and later reference.

## 2. Rotor Blade Model Calculations

The **bldrev.m** subprogram incorporates three major elements: loading the blade and performance information and calculating needed parameters, determining values for the unknown tip displacement and slope, and determining the displacement, slope, moment and shear distributions for the blade. The loading and parameter determination element starts by loading information from the respective parameter and performance data files. The radius of the midpoint of each blade element and the incremental change in radius between elements is then determined. The weight of the blade element is converted into mass and concentrated at the element midpoints (nodes). The thrust and drag forces are broken into steady and harmonic components as discussed in Chapter One. As an aside, **bldrev.m** also contains a diagnostics section that allows for forcing the blade model with a pure harmonic force input. This diagnostic section is accessed manually by editing the MATLAB code in the **bldrev.m** file. The tension in the element and the aerodynamic damping in each element are then calculated. Finally, the twist coupling coefficients are calculated for each element.

The second element takes the information provided by the first element and applies it to the modeling equations as developed in **Appendix B**. In this element, the equations are passed through five times for each forcing harmonic (including the steady forcing or zeroth harmonic); once for each of the unknown tip values of slope and deflection in flatwise and edgewise directions and once for the response

associated with force inputs. Passing through the blade equations results in a transfer matrix for each harmonic that includes blade response and forcing terms as functions of the unknown tip values for flatwise and edgewise slope and deflection. The tip values are determined by solving the equation:

$$Ax + b = 0$$

**Equation 3- 3**

where  $b$  is a vector consisting of the first column of the transfer matrix and representing the forcing terms and where  $A$  is a matrix comprised of the remaining four columns of the transfer matrix representing the blade response terms. The  $A$  and  $b$  terms are reduced to only the row entries needed to satisfy the boundary conditions applicable to the rotor blade root attachment condition, resulting in a four-by-four matrix and a one-by-four vector respectively. In the case of the articulated blade, the boundary conditions require the moment and displacement at the blade root be zero. In the rigid blade root case, the boundary conditions require displacement and slope of the blade be zero at the root. **Equation 3-3** is then used to solve for the values of tip displacement and slope. An example of **Equation 3-3** applied to the  $A$  matrix formed by reducing the entries to those required for the articulated root with lag damper condition is as follows:

$$\begin{bmatrix} M_0^F \\ Z_0 \\ M_0^E - jC_{\theta}\omega\theta_0^E \\ X_0 \end{bmatrix} = [A]^{4 \times 4} \begin{bmatrix} Z_T \\ \theta_T^F \\ X_T \\ \theta_T^E \end{bmatrix} + [b]^{1 \times 4}$$

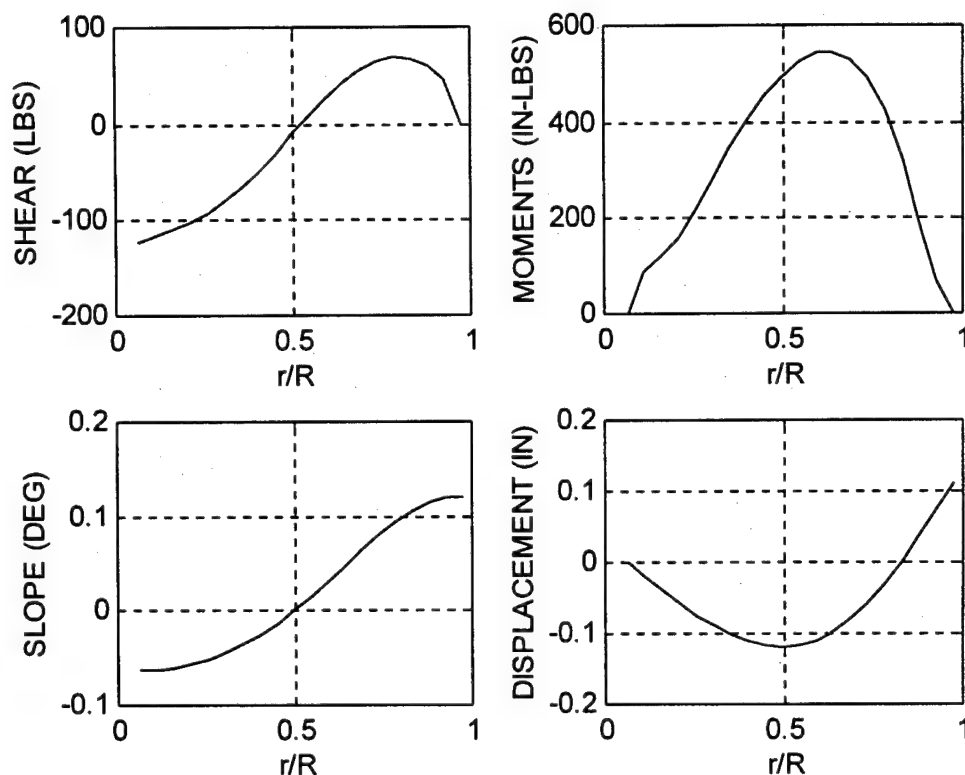
**Equation 3- 4**

In the third element of **bldrev.m**, the previously determined values for the tip slope and displacement are used to initialize the blade equations and pass through the elements again. The resulting displacement, slope, moment and shear values at each node are stored in corresponding matrices by radial position and harmonic. The resulting matrices represent the blade response at each node for each applied harmonic. The matrices are then transferred to the output subprogram to be reassembled as required by the user.

### 3. Rotor Blade Model Outputs Available

The results of the **bldrev.m** calculations are made available to the user through the output subprogram called **output.m**. The **output.m** routine prompts the user to select one of five different formats for the results. The first format presents the displacement, slope, moment and shear distributions

as four distribution plots arranged in 11 sets. These 11 sets of plots correspond to the response of the blade to the individual application of the steady and first ten harmonic forces. In other words, the 11 plot sets depict the rotor blade's response to  $n/rev$  aerodynamic forcing where  $n = 0, 1, 2, \dots, 10$ . **Figure 3-3** illustrates blade flatwise response using the first format for a uniform, articulated rotor blade subjected to the third harmonic component of the forward flight aerodynamic loading. **Figure 3-4** illustrates the response of the same blade subjected to the fourth harmonic of the aerodynamic loading.



**Figure 3-3** Blade flatwise response for a uniform blade under third harmonic loading.

In **Figure 3-3**, the displacement distribution for the uniform blade indicates that the first bending mode of the blade dominates the response to third harmonic (3/rev) loading. The magnitudes of the moment and shear forces are still significant for this load harmonic, even though the displacement of the blade is negligible. The displacement distribution depicted in **Figure 3-4** shows the blade response transitioning from the first to the second bending mode under fourth harmonic (4/rev) loading, with the second mode dominating. The moment and shear plots in **Figure 3-4** indicate that these internal forces become insignificant in magnitude rapidly as the loading harmonic increases.

The trend of the blade responses can be predicted by observing the relationship between the forcing harmonics and the blade natural frequencies as determined in the free vibration case. Under forced vibrations, the blade responds in all of its discrete modes at the frequency of the excitation force. This means that the blade response, although composed of all the individual blade modes, is dominated by the mode closest to the excitation frequency.

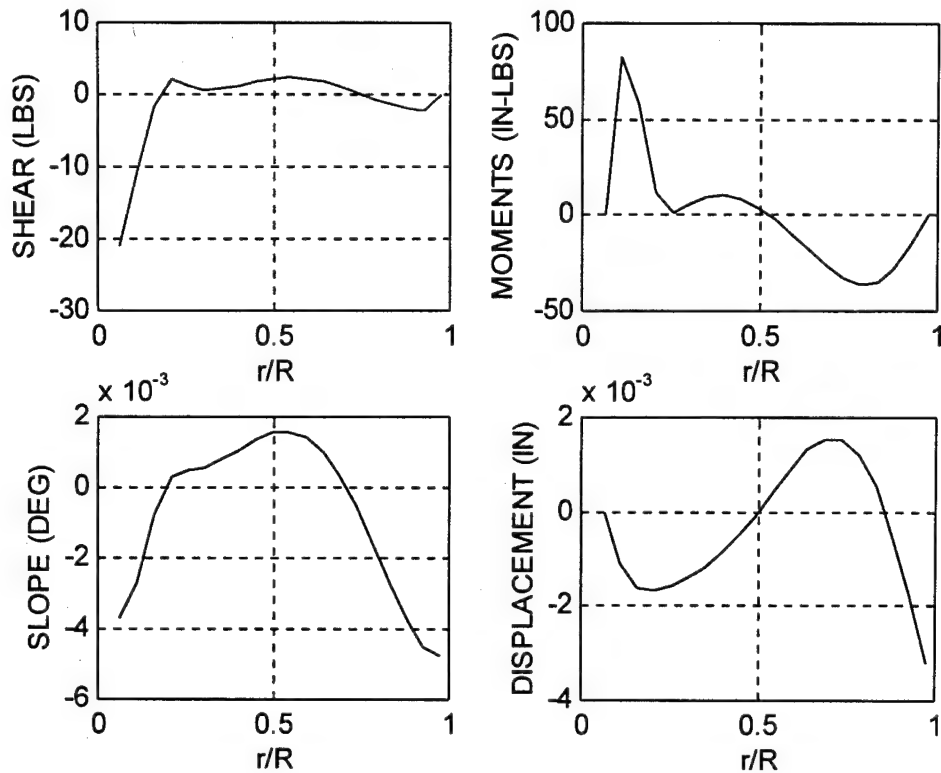


Figure 3- 4 Blade flatwise response for a uniform blade under fourth harmonic loading.

The second format available to view the **bldrev.m** results consists of the displacement, slope, moment and shear plotted as a three-dimensional mesh of total response magnitude versus radial position and azimuth angle. The total blade response at each azimuth position is determined by summing the contribution of each harmonic. **Output.m** performs the calculations necessary to sum the contribution of each harmonic component; essentially reversing the procedure used to harmonically decompose the lift and drag forces. **Figures 3-5 through 3-8** are example plots of the total blade flatwise response distributions for the same uniform blade subjected to the same forward flight aerodynamic loading. The blade response is normally presented as a single plot consisting of four subplots corresponding to displacement, slope, moment and shear, but have been separated for clarity.

Figures 3-7,8 are plots illustrating the moment and shear distributions along the blade for each azimuth position. The shear and moment are zero at the tip of the blade as required by the boundary conditions on the blade. The moment at the blade root and the displacement (Figure 3-5) at the root are also zero as required by the root boundary conditions for an articulated rotor blade. The displacement plot indicates that the blade response is dominated by the steady coning and first harmonic loading as expected from the decomposition of the aerodynamic loading and also by the blade response to individual load harmonics. Figures 3-5 through 3-8 allow the user to visualize the motion and changing internal forces as the blade travels around one complete revolution. Although this format is excellent for visualization of the nature of the blade response, it is not suitable for determining the magnitude of the response. To provide for closer analysis of the blade response, the blade response is cross-sectioned by radial position and by azimuth.

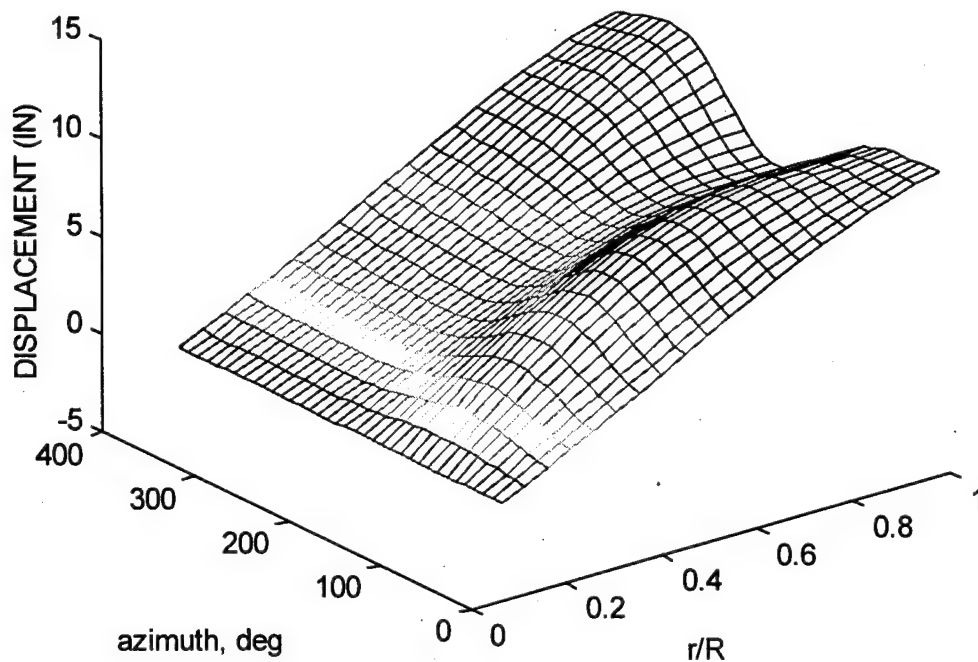


Figure 3- 5 Total blade deflection response for a uniform, articulated blade.

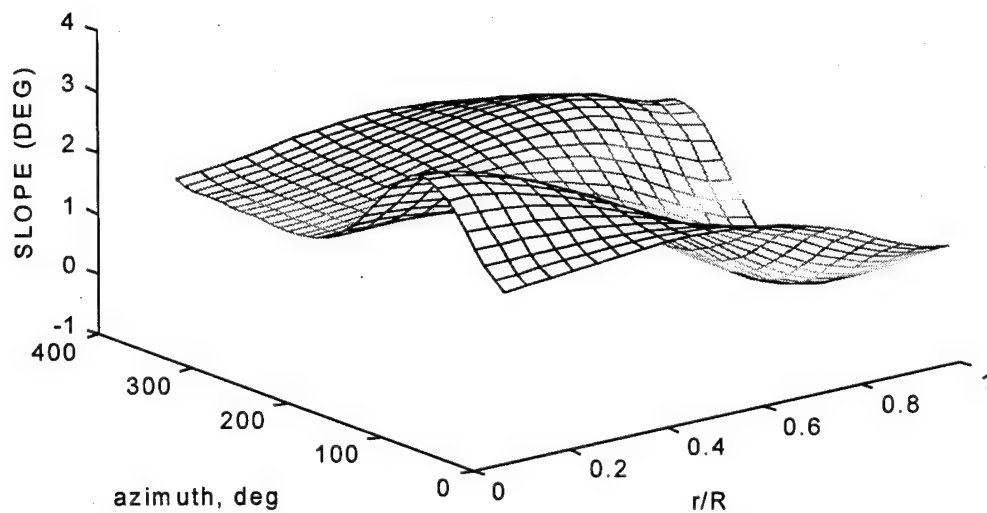


Figure 3- 6 Total blade slope response for a uniform, articulated blade.

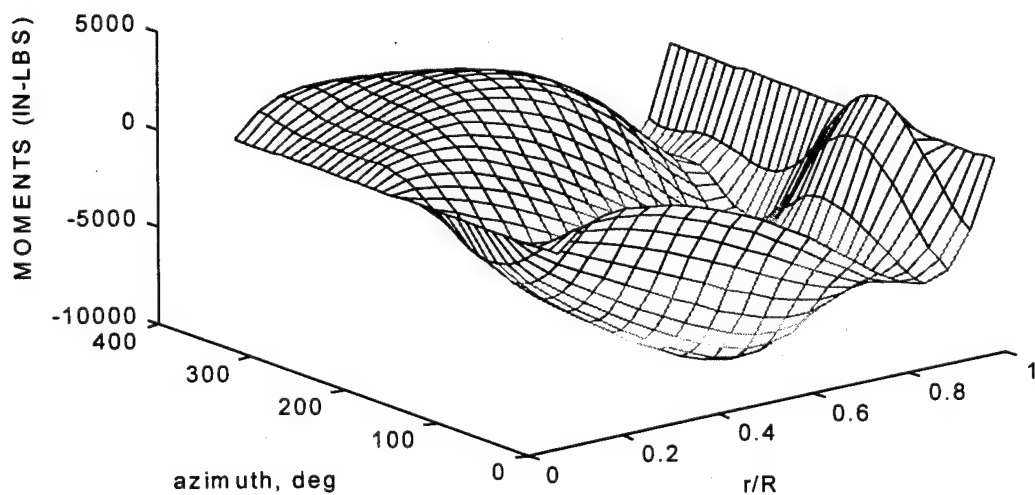
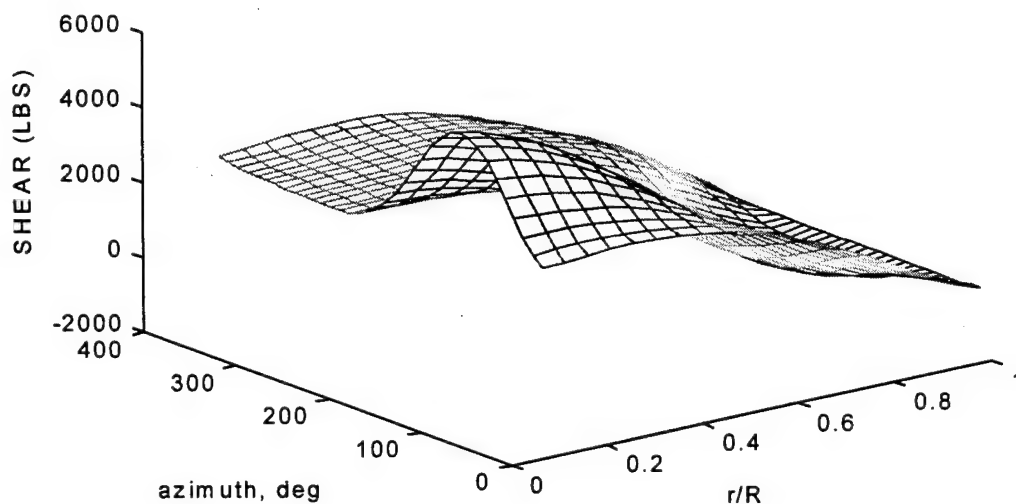
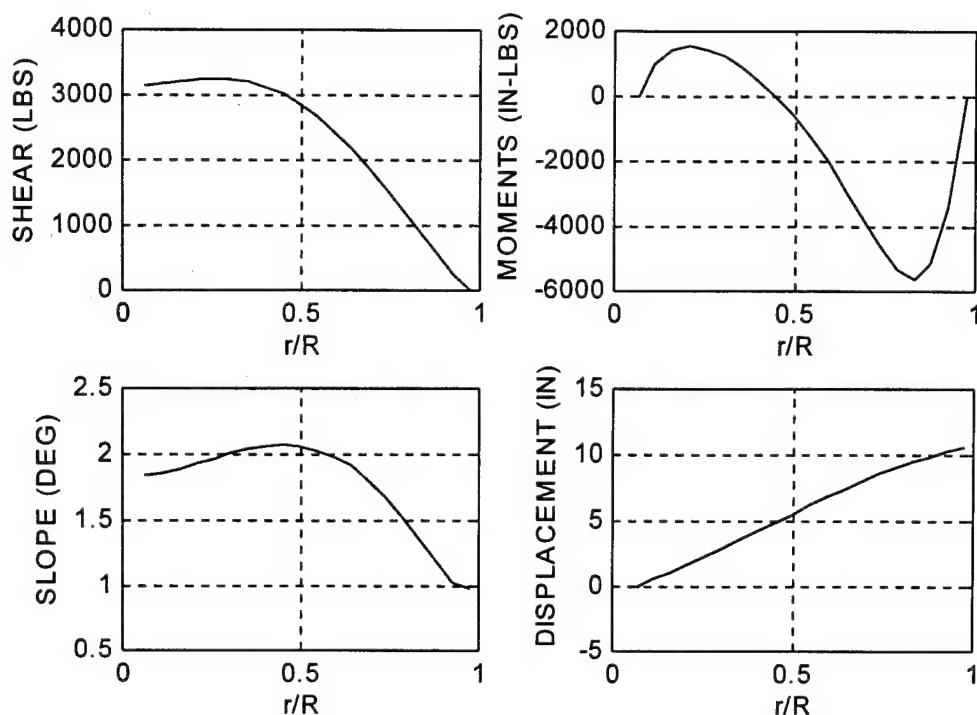


Figure 3- 7 Total blade moment response for a uniform, articulated blade.



**Figure 3- 8 Total blade shear response for a uniform, articulated blade.**

The third format available for presenting the forced blade model results is a composite plot of four subplots corresponding to the displacement, slope, moment and shear distributions for the rotor blade at a specific azimuth position. The `output.m` routine prompts the user to input a desired azimuth for presentation. This means that the general blade response depicted using the three-dimensional mesh plot can be inspected at discrete azimuths. Although the user input a finite number of azimuth stations to perform the response calculations, the azimuth response format will generate an estimated response at an infinite number of azimuth positions. The accuracy of these estimations is directly related to the original number of azimuth stations used to perform the calculations. **Figure 3-9** shows an example of the azimuth response format for the same uniform, articulated blade and steady, inflight loading at zero degrees azimuth. The azimuth response format is useful for determining the limits of the motion and internal forces for a rotor blade subjected to a specific loading condition.



**Figure 3- 9 Total blade response at zero degrees azimuth.**

The fourth format for the blade response is the total response at a specific radial position. This radial position format plots the magnitude of the response versus the azimuth position of the blade for a given radial station. The radial position format is useful for observing the blade response at a specific point, such as at a critical structural point. However, the radial stations available are limited to those corresponding to nodes. This limitation exists because the modeling equations do not resolve the blade response between nodes. The routine accepts any radial position as an input, but the resulting plot will correspond to the blade response at the nearest node. **Figure 3-10** is an example of the radial position format for a uniform, articulated rotor blade subjected to steady, inflight loading. The radial position selected is 0.6 times the rotor radius, which the routine resolves to the node closest to this value. For the example rotor blade with radius  $R$  equal to 31 feet and 20 blade elements, the  $0.6R$  position corresponding to a radius of 18.6 feet has been resolved to the node located at 18.36 feet. Under these rotor radius and number of element conditions, the maximum distance between nodes is 1.5 feet. Therefore, the radial position plotted will be within 0.75 feet of the position requested. Generally, more elements are needed to refine the estimation of the response at a given radial position unless it happens to fall on a node.



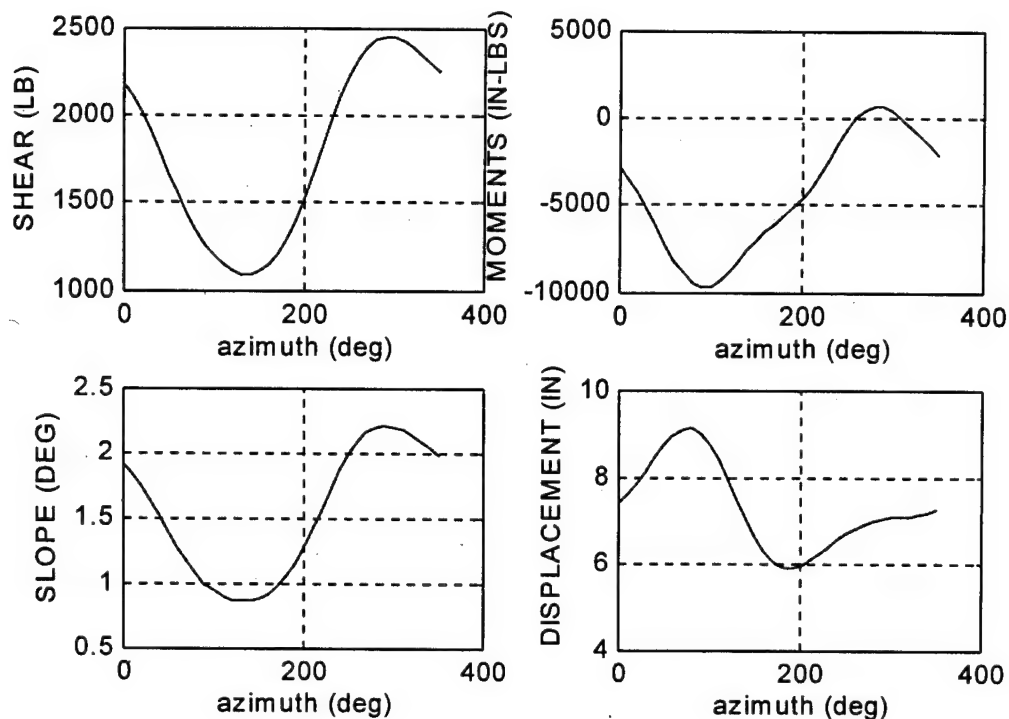


Figure 3- 10 Total blade response at 0.6 R position.

The fifth format available to the user presents a set of two plots corresponding to the blade elastic stiffness and mass distributions. This format provides the user with a visual display of key blade physical parameters, from which the user can determine if the number of blade elements in the model is sufficient to approximate the physical attributes of the actual blade. Figure 3-11 is an example of the blade parameter plots for an actual blade. This example blade is nearly uniform in its elastic stiffness and weight properties, except near the root as required by structural strength needs. This type of blade can be satisfactorily represented by fewer elements than the example blade of Figure 2-9.

All five output formats are designed to provide the user with the minimum tools required to perform preliminary design and analysis of the forced response of common rotor blades. These formats are available for both the flatwise and edgewise motion of the rotor blade. Once the calculation results from `bldrev.m` are made available to the output routine, the user may select any format from the menu and produce hard copies of the resulting plots according to local MATLAB printing procedures.

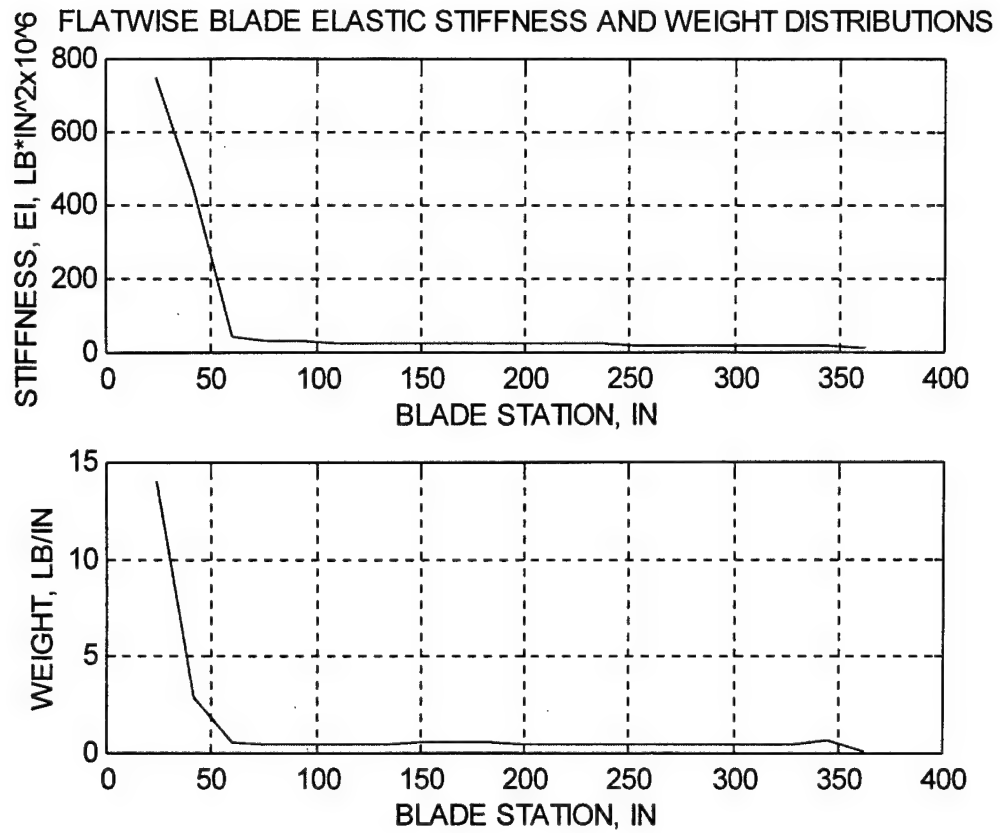


Figure 3- 11 Example blade elastic stiffness and weight distributions

## IV. ROTOR SYSTEM FORCED RESPONSE AND STABILITY

### A. ROTOR FLAPPING STABILITY

#### 1. Rotor Blade Flapping Equation of Motion

In the previous sections, the nature of a rotor blade undergoing free and forced vibration was discussed. Additionally, some tools for approximating the blade's natural frequencies of vibration and its response to steady state aerodynamic loading were developed. One shortcoming of the approximate models developed is that they represent steady state conditions. This means that the models do not give any indication as to the basic stability of the rotor system since they assume that the system is inherently stable. To develop insight into a rotor system's susceptibility to self-exciting vibration forces under certain operating conditions requires a new set of models based on some simplifying assumptions.

In forward flight, the rotor system is susceptible to instability in its flapping motion, given the right combination of flight conditions and rotor physical parameters. Developing a model for analyzing a rotor system design for regions in which these unstable combinations are possible requires that the rotor system be reduced to its simplest form. If the rotor blade is assumed to undergo only rigid body motion, the complexity of the elastic motion of the blade is removed without changing the nature of the blade's overall behavior. To simplify the determination of the flapping moment of inertia for the blade, the rotor blade is assumed to have an articulated root condition. The flapping motion of the blade is assumed to be uncoupled from the pitch and lag motion. Assuming the rotor hub to be fixed in space removes the effects of coupling between the motion of the blade and the motion of the hub and helicopter fuselage. The rotor system reduces to a model of a single rotor blade developed in a rotating coordinate frame. This rotor system model has a single degree of freedom since the motion of all the blades is independent.

The flapping equation of motion for a rotor blade in forward flight is non-linear with time varying coefficients. The equation is non-linear because the blade motion is primarily rotational and has time varying coefficients because the blade experiences changing aerodynamic conditions around the azimuth. Assuming small displacements, the blade motion may be considered linear but the time dependence of the coefficients remains. The rotor blade flapping equation of motion in rotating coordinates as developed by Johnson (1972) is as follows:

$$\ddot{\beta} + \gamma \left[ (K_R M_\theta - M_{\dot{\beta}}) \dot{\beta} + \left( K_P M_\theta - M_\beta + \frac{v^2}{\gamma} \right) \beta \right] = 0$$

Equation 4- 1

In **Equation 4-1**,  $M_0$ ,  $M_{\beta'}$  and  $M_{\beta}$  are blade aerodynamic forces which are dependent on time, and also on the advance ratio of the blade. Since this time dependence is periodic in nature with period  $T$ , the values of the coefficients return to their initial values at the end of one period, i.e.  $A(t+T)=A(t)$ . In the case of a rotor blade, the period  $T$  is equal to the time required for the blade to complete one revolution, i.e. the time for the azimuth angle to change from zero to  $2\pi$  radians. The expressions for  $M_0$ ,  $M_{\beta'}$  and  $M_{\beta}$  change according to the flight regime the rotor system is operating in and the azimuth position of the blade. The disk defining all rotor blade azimuth positions is divided into three regions defined by the following relations:

region (1)  $0 < \mu \sin \psi < \mu$

region (2)  $-1 < \mu \sin \psi < 0$

region (3)  $-\mu < \mu \sin \psi < -1$

These regions are assigned according to the aerodynamic conditions that prevail within them. In region (1), the air flow over the blade span is normal. In region (2), an area of reverse flow exists inboard of the blade radial position  $r = -\mu \sin \psi$  with normal flow outboard of this point. Region (3) is only encountered when the advance ratio  $\mu$  exceeds unity. In region (3), the blade is subjected to reverse flow conditions over the entire span. In the case of hovering flight, where  $\mu = 0$ , the entire disk operates under the normal flow conditions of region (1). As the advance ratio increases, the influence of reverse flow increases and thus the size of regions (2) and (3) also increases. Without including the effects of reverse flow, the blade flapping expression is only valid to an advance ratio equal to 0.5. **Table 4-1** lists the expressions for  $M_0$ ,  $M_{\beta'}$  and  $M_{\beta}$  according to regions.

	region (1)	region (2)	region (3)
$M_{\beta'}$	$-(1/8 + \mu \sin \psi / 6)$	$-(1/8 + \mu \sin \psi / 6 + (\mu \sin \psi)^4 / 12)$	$1/8 + \mu \sin \psi / 6$
$M_{\beta}$	$-\mu \cos \psi (1/6 + \mu \sin \psi / 4)$	$-\mu \cos \psi (1/6 + \mu \sin \psi / 4 - (\mu \sin \psi)^3 / 6)$	$\mu \cos \psi (1/6 + \mu \sin \psi / 4)$
$M_0$	$1/8 + \mu \sin \psi / 3 + (\mu \sin \psi)^2 / 4$	$1/8 + \mu \sin \psi / 3 + (\mu \sin \psi)^2 / 4 - (\mu \sin \psi)^4 / 12$	$-1/8 - \mu \sin \psi / 3 - (\mu \sin \psi)^2 / 4$

**Table 4- 1 Expressions for blade flapping equation coefficients according to region.**

The terms  $K_P$  and  $K_R$ , included in **Equation 4-1**, represent flap proportional feedback gain and flap rate feedback gain respectively.  $K_P$ , defined as the tangent of the pitch-flap coupling angle provided by either blade flap hinge or control system geometry, relates to a physical parameter of the rotor blade known as delta-three. Both feedback terms are included so that the blade pitch change  $\Delta\theta$  due to flapping feedback can be controlled according to some law, such as  $\Delta\theta = -K_P \beta - K_R \dot{\beta}$ . The change in pitch due to

flapping motion can be used to either increase or decrease the forces producing flapping motion by increasing or decreasing the blade pitch angle as the blade flaps. The inclusion of feedback control provides one method by which the rotor blade stability can be affected without changing the physical parameters of the blade itself.

## 2. Rotor Flapping Stability Model Software

Information about the stability of the rotor system in flapping motion is contained in the eigenvalues of **Equation 4-1**. The eigenvalues determine the rotor system natural vibration frequencies and the level of damping present. A stable rotor system has all the eigenvalues of the flapping equation in the left-half coordinate plane, i.e. all the damping values are negative. The system becomes unstable when these values for damping become non-negative. At this point the rotor system is susceptible to self-exciting vibration forces, where energy from another source amplifies the naturally occurring oscillations without bound.

The eigenvalues for the blade flapping equation change throughout the azimuth because the coefficients of the blade flapping equation are a function of the blade azimuth. Extracting meaningful eigenvalues from **Equation 4-1** requires the use of Floquet theory, as described in **Appendix B**. Floquet theory provides a means by which all the information about the eigenvalues for one full revolution of the blade is determined from a single equation. By integrating **Equation 4-1** over one revolution, the resulting expression contains the eigenvalue information for not only one revolution, but for any number of revolutions. Therefore, the stability of the blade in flapping motion, once determined for a given set of conditions, is valid for as long as these conditions remain in effect.

In order to apply Floquet theory to the blade flapping problem, **Equation 4-1** is rewritten in state-space form as:

$$\begin{bmatrix} \dot{\beta} \\ \ddot{\beta} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \gamma \left( M_{\beta} - K_P M_{\theta} - \frac{v^2}{\gamma} \right) & \gamma \left( M_{\dot{\beta}} - K_R M_{\theta} \right) \end{bmatrix} \begin{bmatrix} \beta \\ \dot{\beta} \end{bmatrix}$$

**Equation 4-2**

**Equation 4-2** is integrated using a numerical integration routine such as the fourth order Runge-Kutta method over the period from zero to  $2\pi$ . Once the integration is complete, the state-space matrix corresponds to the state transition matrix  $\alpha$  of **Appendix B** and contains all the information for one complete revolution of the blade. The state transition matrix yields the eigenvalues for the rotor system from which the natural frequency and damping information is extracted.

The MATLAB routines **advfloq.m** and **coeff.m**, developed by the author, execute the principles of Floquet theory for the rotor blade flapping equation. **Advfloq.m** is the main program that performs fourth order Runge-Kutta integration on the state-space matrix of **Equation 4-2**. **Coeff.m** is a function called by **advfloq.m** which returns values for the coefficients  $M_\theta$ ,  $M_\beta$ , and  $M_\beta$  based upon the region the rotor blade is currently operating in. **Advfloq.m** performs the integration over period  $T = 2\pi$  for each of the initial conditions of  $\beta = 1$  and  $\beta' = 1$ . The eigenvalues  $\Theta$  of the resulting  $\alpha$  matrix are determined and then transformed into the corresponding eigenvalues  $\Lambda$  of the  $\beta$  matrix of **Appendix B**. The rotor system damping and frequency are related to  $\Lambda$  according to:

$$\zeta = \text{real}(\Lambda) \quad \text{and} \quad \omega = \text{imaginary}(\Lambda)$$

**Equation 4- 3**

where  $\zeta$  represents the system damping and  $\omega$  represents the natural frequency. The **advfloq.m** routine performs the integration process for a number of advance ratio values and returns plots of the damping and frequency values as a function of advance ratio.

The **advfloq.m** program requires values for the rotor blade Lock number  $\gamma$ , the blade flapping natural frequency ratio  $\nu$ , the pitch-flap coupling angle  $\delta_3$ , the flap rate feedback gain  $K_R$  and the advance ratio. The Lock number, which represents the ratio of aerodynamic forces to inertial forces on the rotor blade, assumes a constant-chord, untwisted blade and is given by the equation:

$$\gamma = \rho \frac{dC_l}{d\alpha} c \frac{R^4}{I_b}$$

**Equation 4- 4**

$$\nu = \sqrt{1 + \frac{eS_b}{I_b}}$$

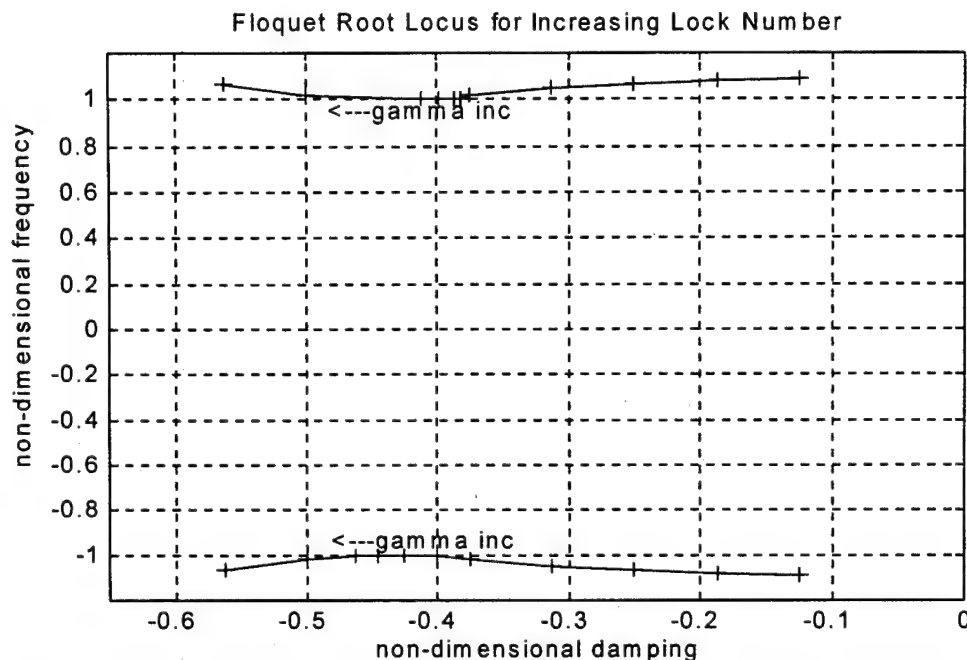
**Equation 4- 5**

The blade flapping natural frequency ratio is determined by the **bldrev.m** subprogram or the equation:

### 3. Software Output

The **advfloq.m** routine provides a means for analyzing the effects of different combinations of rotor physical parameters and control feedback on rotor flapping stability over a range of advance ratios. The program requires editing to modify parameters or the type output available. There are a variety of

ways to present the eigenvalue information determined. One standard presentation is the root locus plot. The root locus plot gives a visual representation of the movement of the eigenvalues, and thus the frequency and damping values, for changes in physical parameters or operating conditions. In **Figure 4-1**, the root locus plot for an example blade using the parameters  $\nu = 1.1$ ,  $\mu = .3$ ,  $K_R = 0$ ,  $K_P = 0$  and  $\gamma = 2, 3, 4, 5, 6, 6.25, 6.5, 6.75, 7, 8, 9$ .

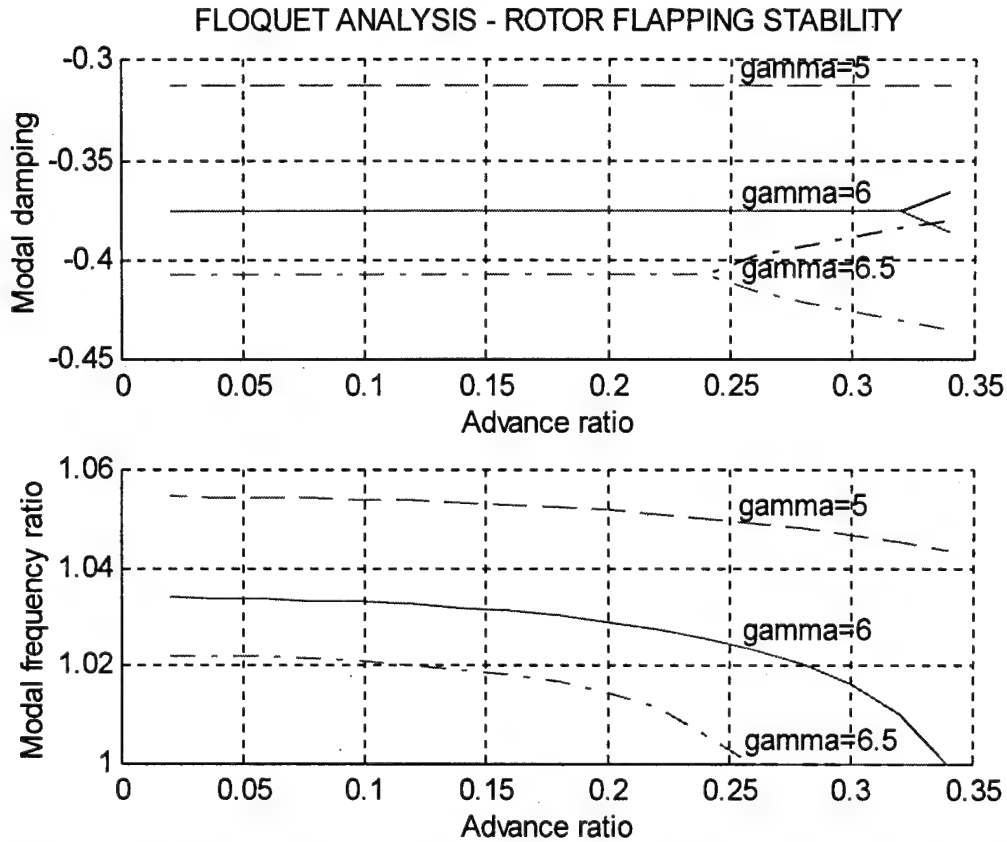


**Figure 4-1** Effects of increasing blade Lock number on rotor stability roots.

**Figure 4-1** illustrates an important attribute of the eigenvalue solution provided by Floquet analysis. As discussed in **Appendix B**, the frequency information cannot be determined exactly from the eigenvalue. The eigenvalues returned by Floquet theory yield frequencies that are accurate to some integer multiple of 0.5/rev. In **Figure 4-1**, the frequency ratio decreases initially to one, stays constant from approximately  $\gamma = 6.25$  to  $\gamma = 7$ , then increases again. As the Lock number increases, the magnitude of the damping is always increasing. The behavior of the frequency and damping values in this example is typical of periodic coefficient systems and directly relates to a constant coefficient system where the root locus of a complex conjugate pair meets the real axis and splits. At this point, the roots are no longer complex nor are they conjugates, with damping increasing in one branch and decreasing in the other. The analogous behavior in a Floquet root locus is the frequency decreasing to some integral multiple of 0.5/rev and becoming stationary. The damping values then separate as in the constant coefficient case. In **Figure 4-1**, the damping values corresponding to  $\gamma = 6.25$  through  $\gamma = 7$  are not conjugates and represent points where the frequency ratio has already stabilized at one and the damping has already separated. The

separation of the damping values indicates critical regions for the blade Lock number where the system is becoming less stable as the magnitude of damping in one branch of the root locus decreases to zero.

To illustrate the Floquet eigenvalue solution behavior from another perspective, a second format for presentation is used. In this format, the damping and frequency values are plotted separately against advance ratio. The Lock number of the blade is held constant while the advance ratio is varied from 0.02 to 0.34. **Figure 4-2** shows the effect of increasing advance ratio for three values of the Lock number.



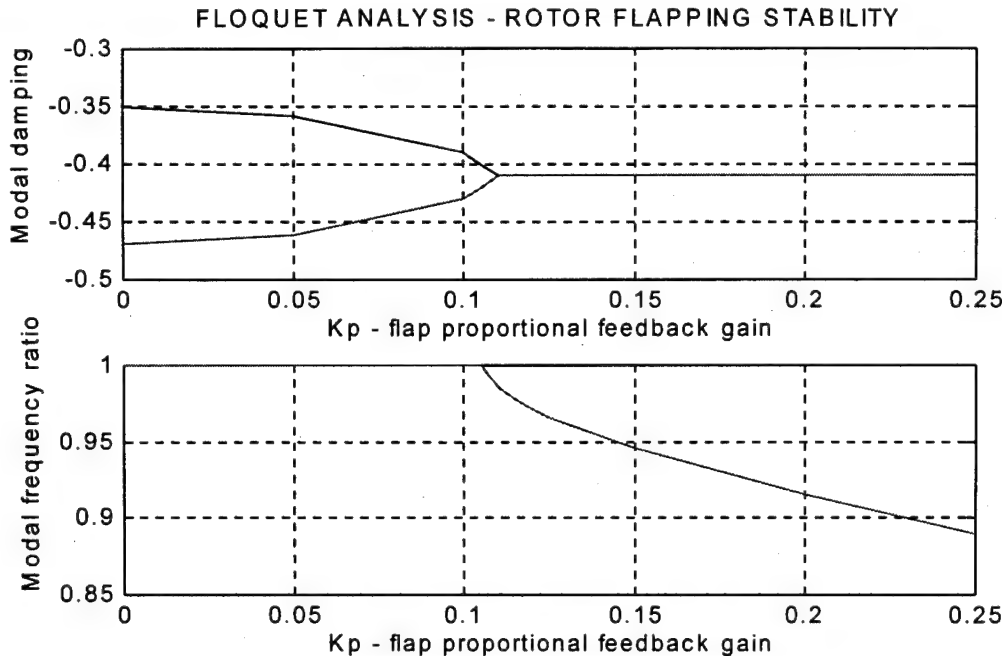
**Figure 4-2** Effect of increasing advance ratio on damping and frequency for three Lock numbers.

**Figure 4-2** confirms the information presented in **Figure 4-1**, where the Lock numbers equal to five and six remain stable at an advance ratio of 0.3, i.e. the frequency is clear of an integral multiple of 0.5/rev. The frequency and damping plots for Lock number equal to 6.5 indicate that the blade enters an unstable region at an advance ratio near 0.25, where the frequency ratio reaches unity and the damping separates into two branches. This confirms the observation from **Figure 4-1** that the frequency ratio stabilizes and the damping separates prior to the advance ratio of 0.3 for this Lock number. A series of plots like **Figure 4-2**, generated for a range of blade Lock numbers, yields a graph showing regions of critical combinations



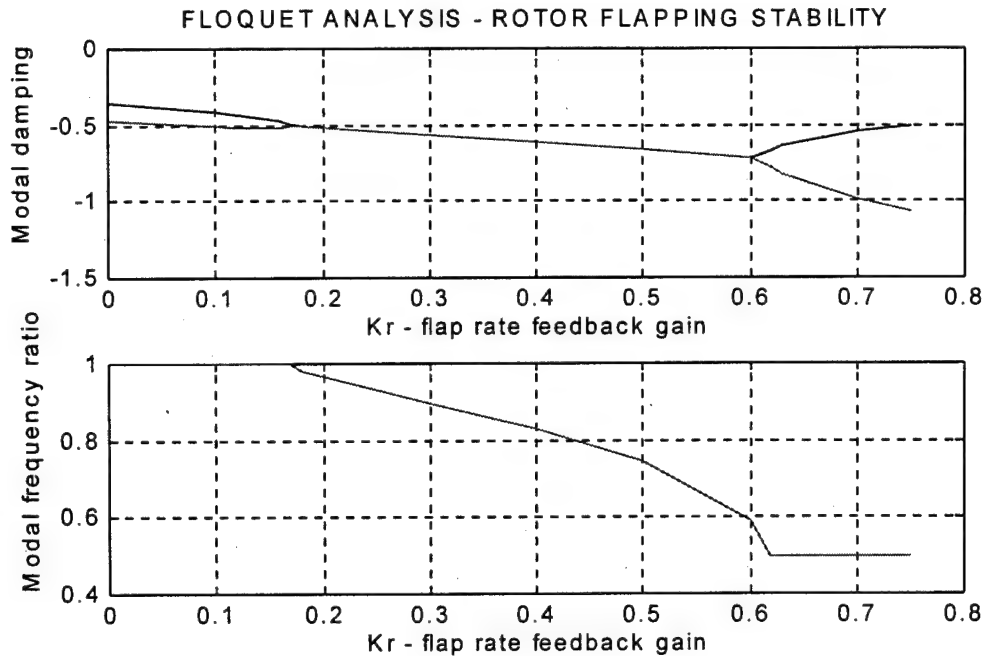
of Lock number and advance ratio. These critical regions include all combinations of Lock number and advance ratio that result in the frequency ratio stabilizing at an integral multiple of 0.5/rev.

The combination of  $\gamma = 6.5$ ,  $\mu = 0.3$  and  $\nu = 1.1$  resides in the 1/rev critical region since the frequency ratio stabilized at 1/rev and the damping values separated. If the rotor system is constrained to this Lock number and flapping natural frequency, but has a requirement to operate up to an advance ratio of 0.5, then feedback control is needed to provide the required performance. Figure 4-3 illustrates the effect of increasing the flap proportional feedback gain  $K_p$ .



**Figure 4-3 Stabilizing effect of increasing flap porportional feedback gain.**

Increasing the flap proportional feedback gain to approximately .105 removes the rotor from the 1/rev critical region. This gain value is equivalent to approximately six degrees of pitch-flap coupling ( $\delta$  three) angle. Increasing the gain value beyond this point serves to increase the margin from the critical region at the cost of increased flap hinge size or control system complexity. If the hinge or control geometry cannot be adjusted to provide the flap proportional feedback gain necessary to stabilize the rotor, the use flap rate feedback control can be considered. In Figure 4-4, the flap rate feedback gain is increased and its effect on the frequency ratio and damping is plotted. The rotor system retreats from the 1/rev critical region as the gain is increased past 0.175 until the gain reaches approximately 0.61. At this point the rotor system enters the 0.5/rev critical region and remains there. Therefore, there exists a range of flap rate feedback gain values that stabilize the rotor and provide margin from both critical regions.



**Figure 4- 4 Effect of flap rate feedback on rotor stability.**

The dynamic flapping stability of the rotor system in forward flight is an important aspect of the helicopter's stability. It provides an indication of the rotor systems ability to achieve high speed flight and to maintain stability when subjected to transient control inputs or aerodynamic gusts. Because the blade motion is a strong function of forward speed, a constant coefficient representation of the system is only adequate in a hover. As the advance ratio increases, the influence of velocity and the importance of reverse flow also increase. The periodic nature of the coefficients of the motion equation are handled by applying Floquet theory. Although it does not provide a solution to the equation of motion, Floquet theory gives insight to the nature of the blade response through eigenvalue analysis. Using Floquet theory, the stability of different combinations of blade physical parameters and advance ratios as well as the effects that feeding back proportional flapping and flap rate has on rotor stability can be investigated.

## **B. AIR AND GROUND RESONANCE STABILITY**

### **1. Rotor System Equations of Motion**

In section A, the stability of the rotor system while in forward flight and subjected to aerodynamic loading is analyzed using Floquet theory. In this section, the effect of coupling on the stability of the system is discussed and a model for analysis is developed. Coupling between the rotor blade lag motion, in-plane motion of the hub and rigid body motion of the airframe can result in a destructive resonant

condition. This condition is most prevalent when the helicopter is in contact with the ground or in a hover (most notably when conducting external cargo operations). The oscillations begin when the center of mass of the rotor blades is displaced from that of the turning rotor hub. The stability of the rotor system is determined from its response to this excitation. The air/ground resonant condition implies that insufficient damping is available in the system to attenuate the oscillations. Without sufficient damping, energy provided by the engines to rotate the hub also goes to increasing the magnitude of the oscillations. Unless the helicopter can be removed from the resonant condition, the vibrations grow without bound and the forces associated with the vibration quickly reach a destructive level.

As in the case of flapping stability, the equations of motion for rotor system lag stability are non-linear with periodic coefficients. The equations linearize around a trim condition assuming small amplitude motions. The periodicity of the coefficients comes, not from aerodynamic effects, but from combining the blade equations developed in a rotating coordinate frame with the hub equations (referring in this case to the combination of the hub itself and the fuselage) developed in a fixed coordinate frame. To remove this periodic nature, the hub equations must be transformed into the rotating coordinate frame with the rotor blades. This transformation requires the hub to have isotropic elastic stiffness and damping characteristics. **Appendix C** develops the basic equation set for a rotor system with any number of rotor blades. As in the flapping stability case, the blade is assumed to undergo rigid body motion only with no pitch or flap motion coupling.

The rotor blade equations of motion assume that the blades are dependent on inter-blade motion and hub motion. This means that even though the hub is assumed isotropic, the individual blades are allowed to have unique values for lag damping. The stability of the rotor system is determined from eigenvalue analysis of the combination of **Equations 9 and 11** in a state-space representation. Standard eigenvalue determination techniques are used since the constant coefficient representation does not require integration of the equations about the azimuth.

## **2. Rotor Lag Stability Model Software**

The MATLAB routine **ccgres.m**, developed by the author and included in **Appendix D**, implements the constant coefficient rotor blade lag equations developed by Hammond (1974). The routine assumes a four bladed rotor system with an isotropic hub. Although the physical parameters of the hub and blades may be edited at will, the number of blades must remain fixed at four unless new coefficient matrices appropriate for the desired number of blades are developed. The entries for the matrices are determined and then placed in position. The resulting state-space representation (reduced for display) of the equations of motion takes the following form for a 4 bladed rotor:

$$\begin{bmatrix} \ddot{\zeta}_{1-4} \\ \ddot{x}_h \\ \ddot{y}_h \\ \dot{\zeta}_{1-4} \\ \dot{x}_h \\ \dot{y}_h \end{bmatrix} = \begin{bmatrix} C_1 & C_2 \\ I & 0 \end{bmatrix}^{12 \times 12} \begin{bmatrix} \dot{\zeta}_{1-4} \\ \dot{x}_h \\ \dot{y}_h \\ \zeta_{1-4} \\ x_h \\ y_h \end{bmatrix}$$

**Equation 4-6**

**Ccgres.m** determines the eigenvalues of **Equation 4-6** for each rotor speed desired and returns a set of plots relating frequency and damping to rotor speed. As in the case of rotor flapping stability, as long as the system damping values remain negative, the rotor system remains stable. Once the damping value in any branch reaches zero, an instability can occur. The model has six degrees of freedom, resulting in six eigenvalues representing six modes of vibration.

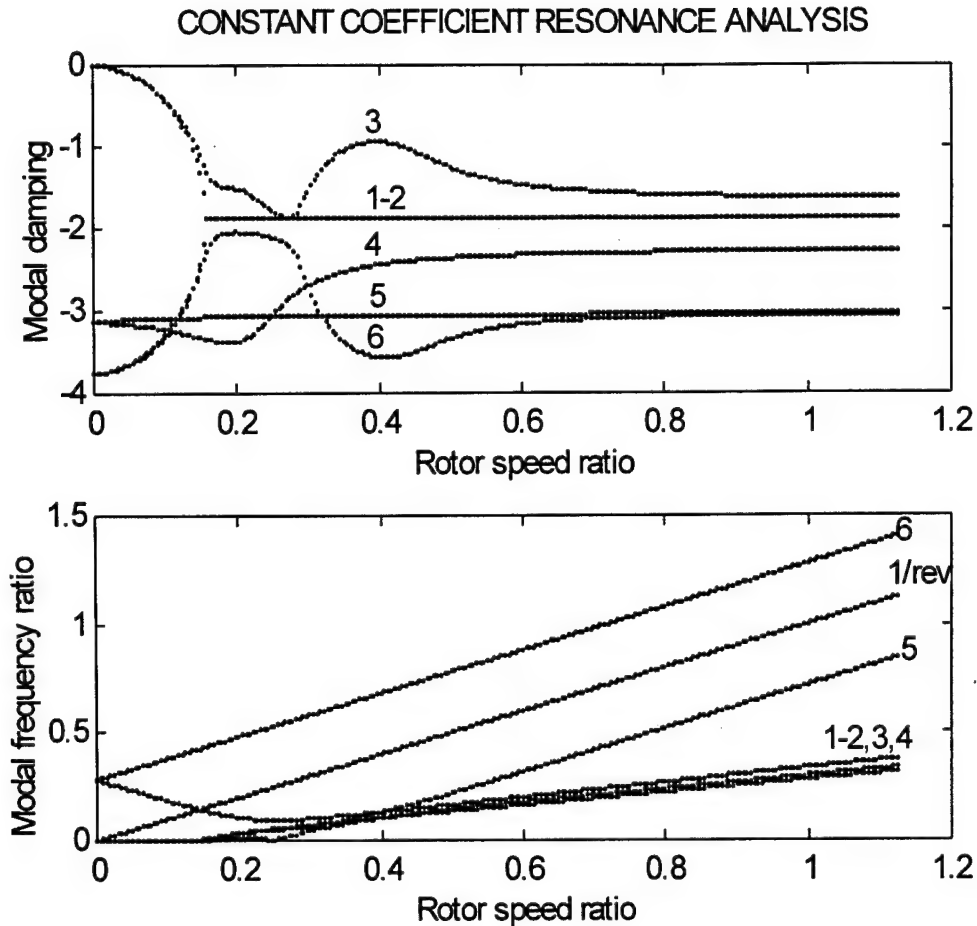
### 3. Software Outputs

The **ccgres.m** routine provides a set of plots for damping and frequency ratio as a function of rotor speed ratio. The example helicopter parameters, taken from Hammond (1974), are as follows:

number of blades	4	$k_i$	0 ft-lb/rad
$m_b$	6.5 slugs	$c_i$	3000 ft-lb-sec/rad
$S_b$	65 slug-ft	$m_x = m_y$	550 slugs
$I_b$	800 slug-ft <sup>2</sup>	$k_x = k_y$	85000 lb/ft
$e$	1 ft	$c_x = c_y$	3500 lb-sec/ft

**Table 4-2 Example helicopter parameters used. After Ref [4]**

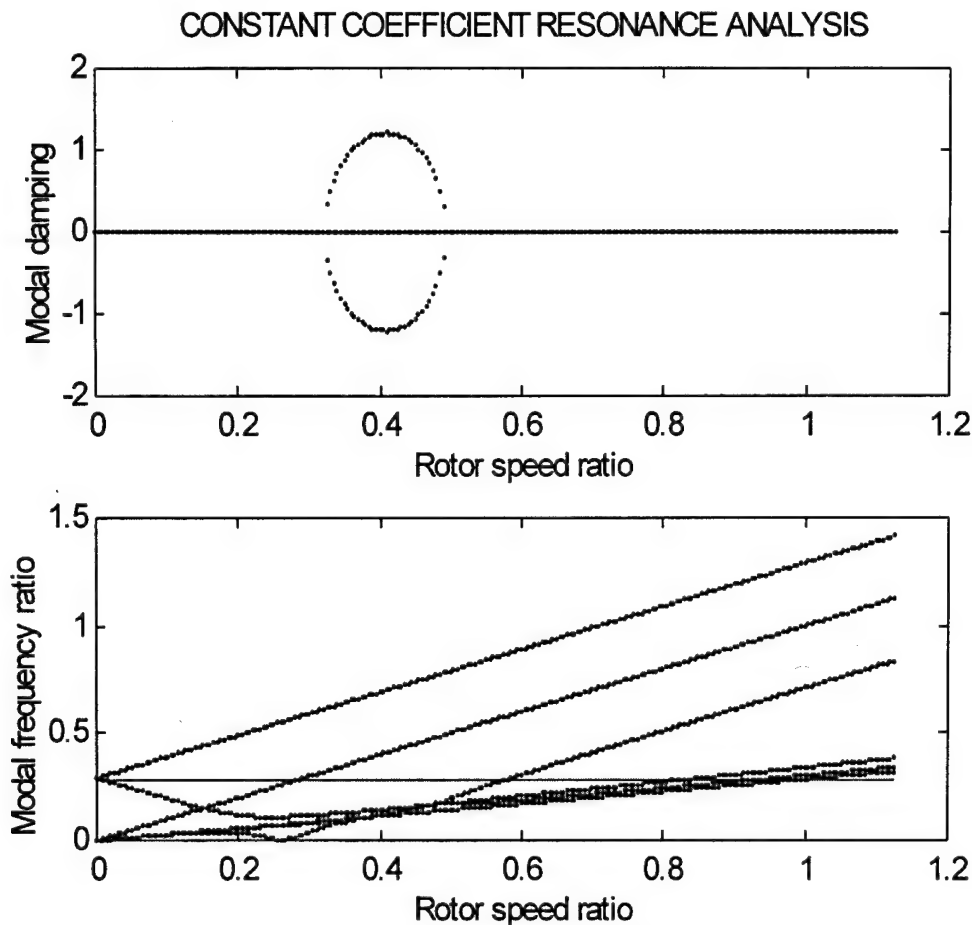
In **Figure 4-5**, the rotor system eigenvalues for rotor speeds from zero to 450 revolutions per minute (rpm) are plotted as damping and frequency ratio versus rotational speed ratio. A rotor normal operating speed of 400 rpm is used to ratio the frequency and rotational speed scales. The 6 modes are arbitrarily labeled in the order of increasing modal frequency ratio.



**Figure 4- 5 Modal damping and frequency ratio for an isotropic rotor system.**

The negative values of modal damping in the damping plot of **Figure 4-5** indicate rotor stability over the entire range of speed ratios. In the present configuration, the rotor system starts from rest and reaches operational speed without encountering an unstable speed range. The lines labeled five and six represent progressive and regressive lag modes corresponding to  $\Omega + v_c$  and  $\Omega - v_c$  respectively; where  $\Omega$  is the rotor speed and  $v_c$  is the lag natural frequency. The lines labeled three and four are rotor modes while the coincident lines one and two represent collective rotor modes. The most important of the lines depicted is the one corresponding to the regressive lag mode frequency, near which resonance instability normally occurs. To illustrate this point, the *cegres.m* routine is run with zero damping values for both the hub and the lag dampers. The resulting plot set, **Figure 4-6**, includes a horizontal line on the frequency ratio plot indicating the uncoupled hub mode. The damping plot depicts neutral stability over the entire rotor speed range, except for the region where the uncoupled hub mode line passes between the 1/rev and regressive lag mode lines. The positive damping values indicate that the rotor system in this condition is

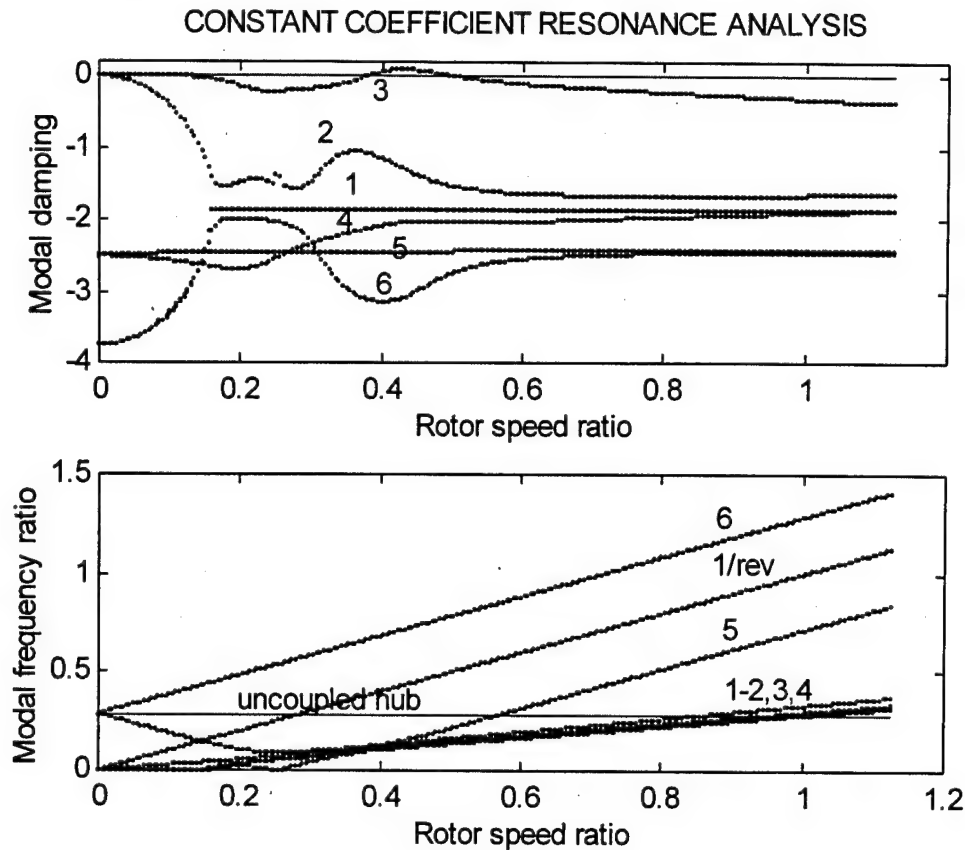
susceptible to self-exciting vibration forces. The region of instability for this helicopter configuration is confined within the bounds of the 1/rev and regressive lag mode lines. The behavior representative of most helicopter configurations is that instability regions, if they exist, are at frequencies less than 1/rev and near the regressive lag mode frequency.



**Figure 4-6** Modal damping and frequency ratio for zero damping conditions.

As previously stated, the `ccgres.m` routine can be used to analyze non-isotropic rotor systems, i.e. the lag damping value for each blade is not the same. A case of particular interest to the United States Army is that of ground operations with one lag damper inoperative. To investigate this condition, the `ccgres.m` program is run with the lag damping coefficient for one blade set to zero. **Figure 4-7** is the plot set for the example helicopter with one damper rendered inoperative. The rotor is unstable in the mode labeled three, which originates from one of the rotor collective modes since the modes labeled one and two are no longer coincident, as they were in **Figure 4-5**. The instability is encountered, as expected, in the

frequency region below the 1/rev line and in the vicinity of the regressive lag mode line along the uncoupled hub mode line.



**Figure 4- 7 Modal damping and frequency ratios for one damper inoperative case.**

The Army design requirements for helicopters state that the helicopter shall be free of resonance with one damper inoperative. In order for the example helicopter to meet this requirement, more damping must be incorporated into the design. Increasing the damping in the other blades will further stabilize all the other modes and eventually stabilize the mode that is currently unstable. Increased damping in the hub or fuselage yields the same results. If the fuselage damping is increased by a factor of two, the unstable mode three becomes neutrally stable. With one damper inoperative, neutral stability is the best condition that can be attained.

When all the blade lag dampers are operative, there exists a minimum value of helicopter damping that ensures stability. This value represents the minimum level of the combined structural, landing gear and rotor blade damping. The damping criteria for stability, given by Deutsch (1946), requires the product of hub and blade damping to follow the relation:

$$C_{\zeta} C_h > \omega_h^2 (1 - \nu_{\zeta}) \frac{NS_b^2}{2\nu_{\zeta}}$$

**Equation 4- 7**

where  $C_{\zeta}$  is the blade lag damper coefficient,  $C_h$  is the effective hub damping coefficient,  $N$  is the number of blades and  $\omega_h$  is the hub natural frequency. For the example helicopter parameters, the product of blade and hub damping must exceed  $2.804 \times 10^6 \text{ lb}^2\text{-sec}^2/\text{rad}$ . If the rotor blade dampers provide damping equivalent to 3000 ft-lb-sec/rad, the hub and fuselage must provide at least 935 lb-sec/ft for the helicopter to meet the stability criteria. The Deutsch stability criteria indicates the lag damping required to provide stability at the resonant point where the regressive lag mode frequency and hub mode frequency coincide. When the hub is anisotropic, there exists a resonant point for each of the hub modes and a minimum damping requirement to stabilize each mode.



## **V. CONCLUSIONS AND RECOMMENDATIONS**

### **A. CONCLUSIONS**

The rotor system is the primary source of vibratory forces as well as a major factor in the stability and control of a helicopter. A qualitative understanding of the forces exciting the rotor system and the response of the rotor to these forces requires the use of simplifying assumptions and basic analytical tools. This thesis presents the Mykelstad and finite element methods for modeling the rotor system in free and forced vibration, discusses the assumptions required to develop the basic tools and presents computer models useful for preliminary design and qualitative analysis.

Additionally, this thesis discusses the importance of lag stability during ground operations and the influence of increasing forward flight speed on flapping stability. The periodic nature of the motion equations for forward flight stability is handled by simplifying the rotor model and applying Floquet theory to handle the periodicity of the coefficients. The resulting computer routine is useful for studying the combinations of rotor parameters and control feedback required to ensure rotor stability at a given advance ratio. Floquet theory is not required in the ground resonance case when the rotor hub is assumed to be isotropic in its properties. This case is handled effectively with a constant coefficient approximation that is useful in visualizing the operating regions where lag instability occurs and the influence of rotor, hub and fuselage damping in stabilizing the system.

### **B. RECOMMENDATIONS**

In order to improve the basic tools presented in this thesis, and consequently upgrade the JANRAD software that incorporates these tools, the computer programs should be expanded to remove some of the simplifying assumptions. The Myklestad based rotor model presented in this thesis can be expanded to include torsional effects and allow for gimbal and teetering rotor root conditions. The out-of-plane Coriolis force should also be incorporated into the model. Each of the aforementioned items represents an upgrade to the basic rotor model. To integrate this model into the helicopter design, the rotor model should be coupled to a fuselage model. The coupling process provides the user with insight into the complexity of rotor and fuselage interaction.

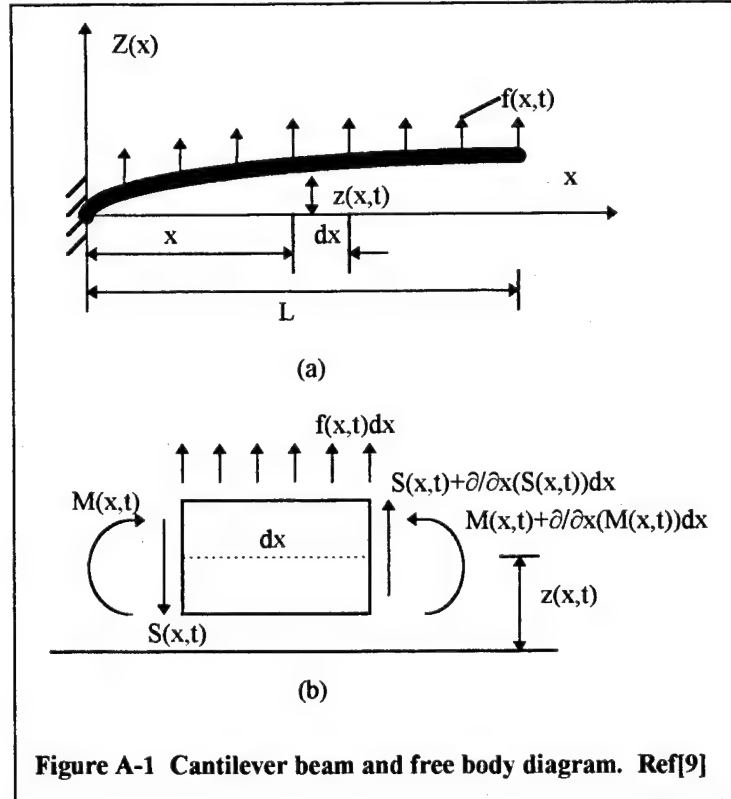
The rotor flapping stability model presented is greatly simplified and, therefore, has a limited utility. The computer program can be adapted to allow a multiblade rotor system with elastic hub effects by following the work of Peters and Hohenemser (1970). Additionally, the assumption of rigid blade motion can be removed to allow low order bending mode influence into the stability model. The resulting

model is especially useful for rotor stability and control analysis at high advance ratios and under the influence of various control feedback schemes.

The ground/air resonance model developed in this thesis is currently limited to isotropic hub conditions, which do not adequately model a real helicopter. Floquet theory, applied to the blade equations in the rotating coordinate frame and the hub equations in the fixed coordinate frame, adds the periodic coefficient effects and allows the inclusion of non-isotropic hub conditions. The resulting model is useful for determining the stability margins of a helicopter design and analyzing off-design conditions such as operations with one damper inoperative and blade-to-blade dissimilarities.

## APPENDIX A. BEAM BENDING VIBRATIONS BY EXACT METHOD

The flexural vibration of a beam is a boundary-value problem defined by the fourth-order differential equation developed from a balance of forces and moments on the cantilever beam element depicted in **Figure A-1**.



From **Figure A-1b**, the equation of motion developed from a force balance on the element is:

$$\left[ S(x,t) + \frac{\partial S(x,t)}{\partial x} dx \right] - S(x,t) + f(x,t)dx = m(x)dx \frac{\partial^2 z(x,t)}{\partial t^2}$$

**Equation A- 1**

The moment balance provides a second equation of motion for the element of the form:

$$\left[ M(x,t) + \frac{\partial M(x,t)}{\partial x} dx \right] - M(x,t) + \left[ S(x,t) + \frac{\partial S(x,t)}{\partial x} dx \right] dx + f(x,t) \frac{(dx)^2}{2} = 0$$

**Equation A- 2**

Canceling like terms in **Equations A-1** and **A-2**, ignoring second order terms and making use of the relations:

$$\frac{\partial M(x,t)}{\partial x} + S(x,t) = 0 \quad \text{and} \quad M(x,t) = EI(x) \frac{\partial^2 z(x,t)}{\partial x^2}$$

**Equation A- 3**

$$-\frac{\partial^2}{\partial x^2} \left[ EI(x) \frac{\partial^2 z(x,t)}{\partial x^2} \right] + f(x,t) = m(x) \frac{\partial^2 z(x,t)}{\partial t^2}$$

**Equation A- 4**

results in the following partial differential equation for the bending vibration of the beam:  
which is valid over the interval  $0 < x < L$ .

If the free vibration case is considered, the forcing term  $f(x,t)$  is equal to zero and the solution to the eigenvalue problem is obtained using:

$$z(x,t) = Z(x)F(t)$$

**Equation A- 5**

where  $Z$  depends on  $x$  only and  $F(t)$  depends only on time and is bounded and harmonic in nature. By using separation of variables, **Equation A-4** can now be rewritten in terms of total derivatives as:

$$\frac{d^2}{dx^2} \left[ EI(x) \frac{d^2 Z(x)}{dx^2} \right] = \omega^2 m(x) Z(x)$$

**Equation A- 6**

An exact solution to **Equation A-6** is possible in the special case of a continuous, uniform beam where the flexural rigidity  $EI(x)$  and the mass per unit length  $m(x)$  are assumed to be constant over the length of the beam. **Equation A-6** reduces to the following form:

$$\frac{d^4 Z(x)}{dx^4} - \beta^4 Z(x) = 0 \quad \text{where} \quad \beta^4 = \frac{\omega^2 m}{EI}$$

**Equation A- 7**

The following boundary conditions at  $x = 0$  and  $x = L$  apply to the uniform cantilever beam of **Figure A-1**:

$$\begin{array}{ll} Z(0) = 0 & \frac{d^2 Z(x)}{dx^2} \Big|_{x=L} = 0 \\ \frac{dZ(x)}{dx} \Big|_{x=0} = 0 & \frac{d^3 Z(x)}{dx^3} \Big|_{x=L} = 0 \end{array}$$

**Equation A- 8**

The general solution of **Equation A-7** is given by:

$$Z(x) = C_1 \sin \beta x + C_2 \cos \beta x + C_3 \sinh \beta x + C_4 \cosh \beta x$$

**Equation A- 9**

Applying the boundary conditions to solve **Equation A-9** for the constants of integration  $C_2 - C_4$  and simplifying the results yields the following equation in terms of  $C_1$  :

$$Z(x) = \frac{C_1}{\sin \beta L - \sinh \beta L} [(\sin \beta L - \sinh \beta L)(\sin \beta x - \sinh \beta x) + (\cos \beta L + \cosh \beta L)(\cos \beta x - \cosh \beta x)]$$

**Equation A- 10**

**Equation A-10** represents the modes of the beam at each value of  $\beta$ , where  $C_1$  is determined by the initial conditions. Since  $C_1 = 0$  only in the trivial case, the expression in the brackets of **Equation A-11** below must equal zero in order to satisfy the boundary conditions of the beam.

$$C_1 [(\sin \beta L + \sinh \beta L)(\sin \beta L - \sinh \beta L) + (\cos \beta L + \cosh \beta L)^2] = 0$$

**Equation A- 11**

**Equation A-11** reduces to the characteristic equation shown below.

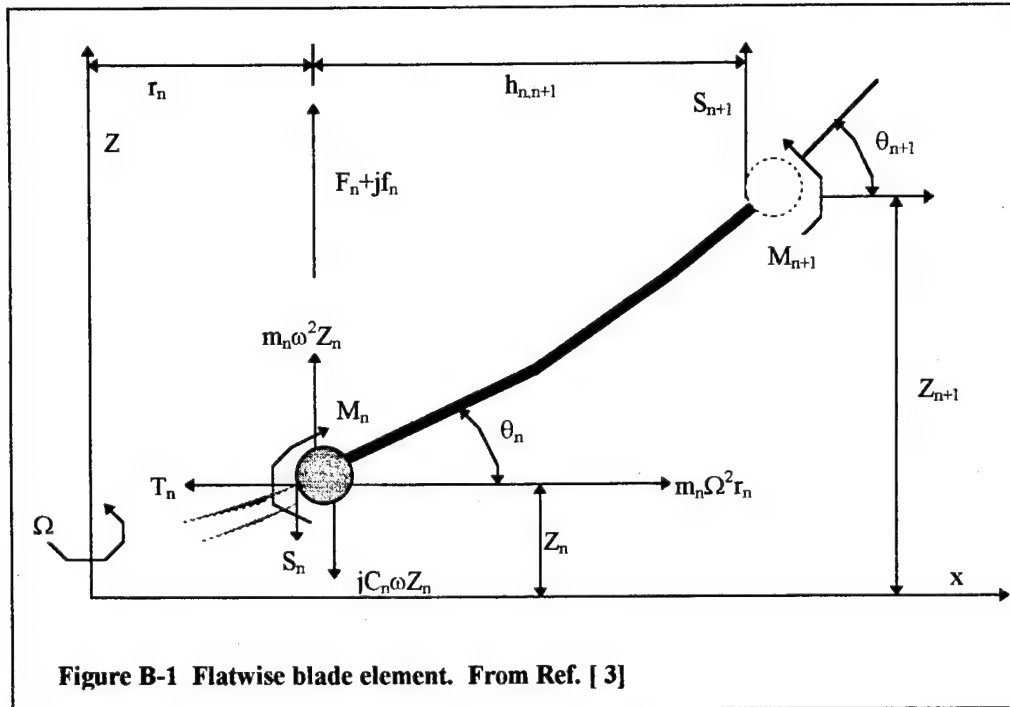
$$\cos(\beta L) = \frac{-1}{\cosh(\beta L)}$$

**Equation A- 12**

**Equation A-12** has an infinite set of solutions for  $\beta$  and consequently for the natural frequency according to the relation in **Equation A-7**. The modes of vibration are determined by inserting into **Equation A-10** the values of  $\beta$  corresponding to each natural frequency and applying the initial conditions for  $C_1$ . A common method of determining an appropriate value for  $C_1$  is to set the deflection of the beam to one unit at  $x = L$  and solving for  $C_1$ . This value may then be used for all other values of  $x$  to plot the mode shape. At higher vibration modes the simple-beam theory on which this method is based is not valid because the rotation of an element of the beam can no longer be considered negligible compared to the translation of the element. The equations derived are applicable to a static, uniform untwisted rotor blade with cantilever root boundary conditions. A more detailed derivation of the exact method is available from numerous sources on vibration analysis. (Meirovitch, 1986)

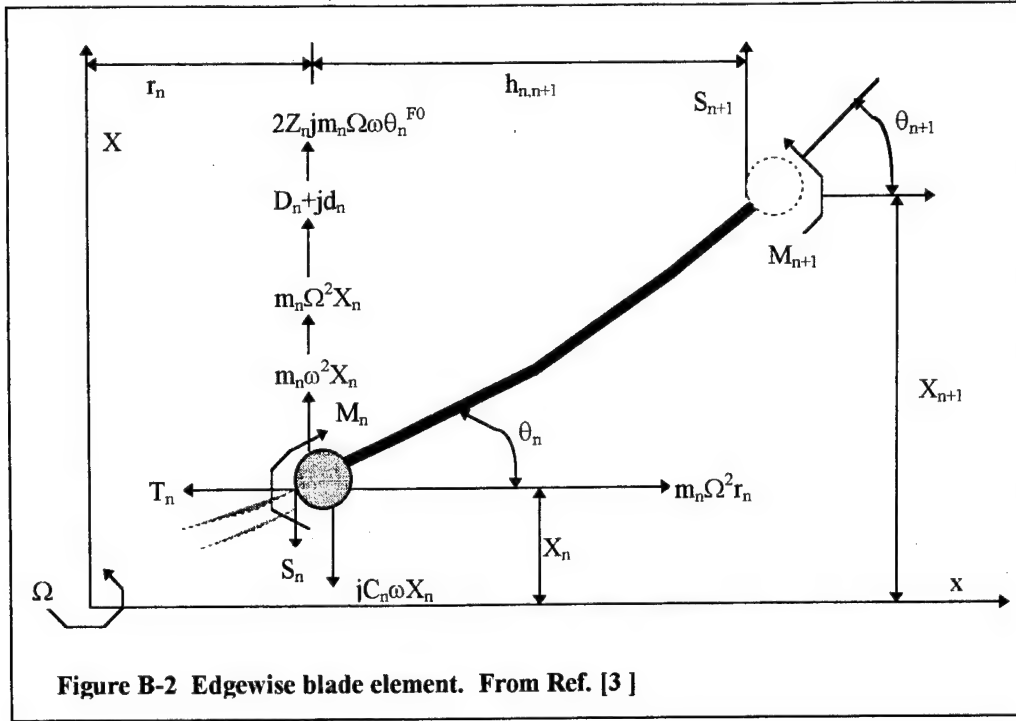
## APPENDIX B. BEAM BENDING VIBRATIONS BY MYKLESTAD'S METHOD

Myklestad's method for flexural beam vibration analysis is based on discretizing a continuous system into a number of rigid concentrated masses connected by massless flexible elements. This discretization results in the replacement of the differential equation of motion for the beam with finite difference equations that relate one element to the next as illustrated in **Figure B-1** and **Figure B-2**.



**Figure B-1** shows the forces and moments acting on a beam element or, in the case of a rotor blade, the forces and moments acting on the rotor blade element under flexural vibration in the flatwise direction. A similar set of forces and moments act on the rotor blade element during edgewise flexural vibration as shown in **Figure B-2**. The solution process involves summing the forces and moments acting on an element and relating one element to the next, starting at the tip of the blade and working to the root.

The flatwise and edgewise element equations are developed in a form suited for rotor blade analysis vice beam analysis since they relate to the rotor blade analysis subprogram developed by the author for the Joint Army/Navy Rotorcraft Analysis and Design (JANRAD) computer program. The centrifugal tension acting on an element is given by:



$$T_n = T_{n+1} + m_n \Omega^2 r_n$$

**Equation B- 1**

The flatwise and edgewise shear forces are given by:

$$\begin{aligned} S_n^F &= S_{n+1}^F + m_n \omega^2 Z_n - j C_n \omega Z_n + F_n + j f_n \\ S_n^E &= S_{n+1}^E + m_n (\omega^2 + \Omega^2) X_n + 2 j m_n \omega \Omega \theta_n^{F0} Z_n + j C_n \omega Z_n + D_n + j d_n \end{aligned}$$

**Equation B- 2a and B-2b**

where superscript  $F$  relates to flatwise components and superscript  $E$  relates to edgewise components. The  $m_n \omega^2 Z_n$  and  $m_n (\omega^2 + \Omega^2) X_n$  terms in **Equations B-2a** and **2b** represent inertial forces acting on the rigid mass of the element. The term  $2 j m_n \omega \Omega \theta_n^{F0} Z_n$  in the edgewise shear equation represents the in-plane Coriolis force produced by the blade out-of-plane flapping motion. The shear equations are expressed in complex form to allow easy application of the harmonic decomposition of lift ( $F_n + j f_n$ ) and drag ( $D_n + j d_n$ ) forces to the element. The  $j C_n \omega Z_n$  term is the blade element's flatwise aerodynamic damping force as given by:



$$C_n = \frac{dC_l}{d\alpha} \text{chord}_n \frac{\rho}{2} \Omega r_n h_{n,n+1}$$

**Equation B- 3**

The flatwise and edgewise equations for moments acting on the element are given by:

$$\begin{aligned} M_n^F &= M_{n+1}^F + S_{n+1}^F h_{n,n+1} - T_{n+1} (Z_{n+1} - Z_n) \\ M_n^E &= M_{n+1}^E + S_{n+1}^E h_{n,n+1} - T_{n+1} (X_{n+1} - X_n) \end{aligned}$$

**Equation B- 4a and B-4b**

The flatwise and edgewise equations for the slope of the element are:

$$\begin{aligned} \theta_n^F &= \theta_{n+1}^F (1 + T_{n+1} u_{zz}) + T_{n+1} \theta_{n+1}^E u_{zx} - M_{n+1}^F v_{zz} - M_{n+1}^E v_{zx} - S_{n+1}^F u_{zz} - S_{n+1}^E u_{zx} \\ \theta_n^E &= \theta_{n+1}^E (1 + T_{n+1} u_{xx}) + T_{n+1} \theta_{n+1}^F u_{xz} - M_{n+1}^E v_{xx} - M_{n+1}^F v_{xz} - S_{n+1}^E u_{xx} - S_{n+1}^F u_{xz} \end{aligned}$$

**Equation B- 5a and B-5b**

The deflection of the element in the flatwise and edgewise directions is given by:

$$\begin{aligned} Z_n &= Z_{n+1} - \theta_n^F h_{n,n+1} + T_{n+1} (\theta_{n+1}^F g_{zz} + \theta_{n+1}^E g_{zx}) - M_{n+1}^F u_{zz} - M_{n+1}^E u_{zx} - S_{n+1}^F g_{zz} - S_{n+1}^E g_{zx} \\ X_n &= X_{n+1} - \theta_n^E h_{n,n+1} + T_{n+1} (\theta_{n+1}^E g_{xx} + \theta_{n+1}^F g_{xz}) - M_{n+1}^E u_{xx} - M_{n+1}^F u_{xz} - S_{n+1}^E g_{xx} - S_{n+1}^F g_{xz} \end{aligned}$$

**Equation B- 6a and B-6b**

The subscripted xx, xz, zz and zx terms in the slope and deflection equations relate the coupling effect of rotor blade twist on the flatwise and edgewise dynamic response of the blade. These terms are more fully developed in the paper by Gerstenberger and Wood (1963).

At the rotor blade tip, the shear, moment and tension in both the flatwise and edgewise directions are zero. The value for the excitation frequency  $\omega$  is assumed to be the same as the frequency of the forcing harmonic applied. The unknown tip values are the slope and deflection terms in flatwise and edgewise directions. The tip slope and deflection terms are carried through as variables in **Equations B-2, B-4, B-5 and B-6** from element to element all the way to the blade root. The result is eight equations in terms of the four unknown tip values of slope and deflection. Four root boundary conditions are then applied to the equations to solve for the unknown tip values. Some of the root boundary conditions of interest are as follows:

$$\theta_0^F = 0 \quad Z_0 = 0 \quad \theta_0^E = 0 \quad X_0 = 0$$

**Equation B- 7 Rigid root boundary conditions**

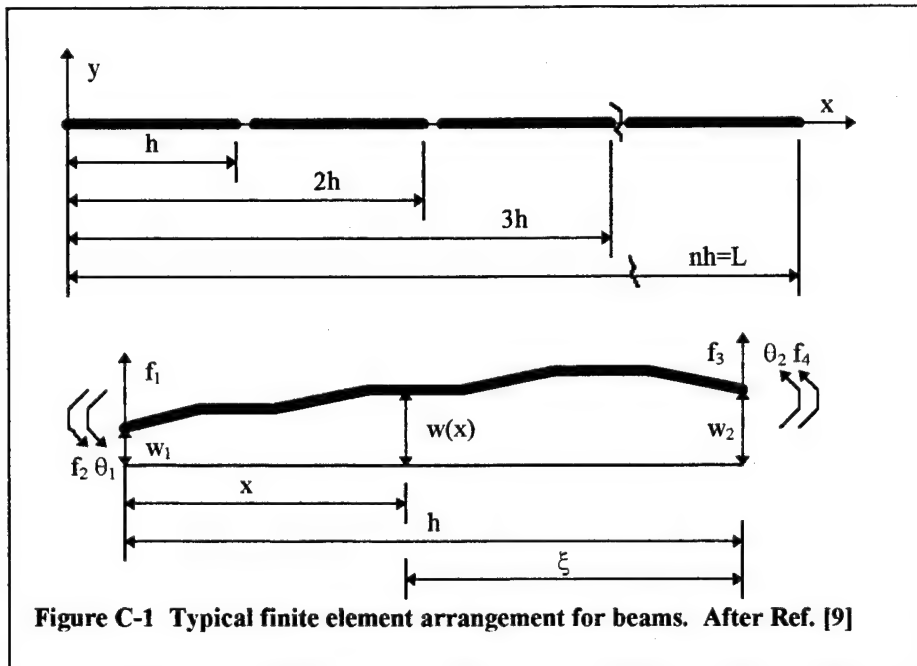
$$M_0^F = 0 \quad Z_0 = 0 \quad M_0^E - jC_0 \omega \theta_0^E = 0 \quad X_0 = 0$$

**Equation B- 8 Articulated root boundary conditions**

The fully defined tip values of slope, deflection, shear and moment are used to determine the response of the entire blade by retracing the steps from element to element with the known values. The blade response is given by the moment, shear, slope and deflection distributions along the length of the rotor blade. This solution process is repeated separately for each forcing harmonic. The total response of the rotor blade is determined by recombining the individual harmonic responses. Additionally, the total response of the blade must be referenced to a specific blade azimuth position since the response will vary with azimuth. (Gerstenberger and Wood, 1963)

## APPENDIX C. BEAM BENDING VIBRATION BY FINITE ELEMENT METHOD

This approximate solution method is considered a special case of the Rayleigh-Ritz method. The essence of the finite element method is a discretization of a continuous system into smaller continuous elements based upon a finite number of degrees of freedom. The displacement of the continuous system is determined by the displacement of a finite number of nodal points and interpolation of these nodal displacements over the length of each element. Each element is tied to the next at the nodal points by the requirement for a balance of forces and compatible displacements at the nodes. A typical beam broken into finite elements and a characteristic element are shown in **Figure C-1**.



**Figure C-1** Typical finite element arrangement for beams. After Ref. [9]

In the development of the element stiffness matrix, **Equation A-6** is rewritten in terms of the element coordinates:

$$\frac{d^2}{dx^2} \left[ EI(x) \frac{d^2 w(x)}{dx^2} \right] = \omega^2 m(x) w(x)$$

**Equation C- 1**

which is valid over the interval  $0 < x < h$ . Assuming no force is applied to the element and the elastic stiffness  $EI(x)$  is constant over the element, **Equation C-1** reduces to the displacement relation:

$$EI \frac{d^4 w(x)}{dx^4} = 0$$

**Equation C- 2**

which is integrated four times to get:

$$w(x) = \frac{1}{6} c_1 x^3 + \frac{1}{2} c_2 x^2 + c_3 x + c_4$$

**Equation C- 3**

The constants of integration are determined by applying the boundary conditions listed below for the element shown in **Figure C-1**.

$$w(0) = w_1 \quad \left. \frac{dw(x)}{dx} \right|_{x=0} = \theta_1 \quad w(h) = w_2 \quad \left. \frac{dw(x)}{dx} \right|_{x=h} = \theta_2$$

**Equation C- 4**

The constants of integration are incorporated into **Equation C-3** to get the following expression for the bending displacement inside the element:

$$w(x) = \left[ 1 - 3\left(\frac{x}{h}\right)^2 + 2\left(\frac{x}{h}\right)^3 \right] w_1 + \left[ \frac{x}{h} - 2\left(\frac{x}{h}\right)^2 + \left(\frac{x}{h}\right)^3 \right] h\theta_1 + \left[ 3\left(\frac{x}{h}\right)^2 - 2\left(\frac{x}{h}\right)^3 \right] w_2 + \left[ -\left(\frac{x}{h}\right)^2 + \left(\frac{x}{h}\right)^3 \right] h\theta_2$$

**Equation C- 5**

**Equation C-5** reduces to :

$$w(x) = L_1(x)w_1 + L_2(x)h\theta_1 + L_3(x)w_2 + L_4(x)h\theta_2$$

**Equation C- 6**

where  $L_i(x)$  ( $i=1,2,3,4$ ) is an interpolating function that determines the displacement at any point inside the element based upon the displacement and slope at the nodes. These interpolating functions represent the lowest order polynomial that can be used to satisfy the fourth-order differential equation of motion.

Higher order polynomials can be added to the basic set of interpolating functions in a hierarchical fashion to increase the accuracy of the approximation. Each added polynomial must have zero amplitude and slope at each node of the element to which it is added since the new polynomial does not represent a

physical state as such. A suitable form for a hierarchical function can be generated from the following equation:

$$L_{4+i} = \xi^2 (1-\xi)^2 \prod_{j=2}^i (j-1-i\xi)$$

**Equation C- 7**

Each new function added increases the order of the mass and stiffness matrices by one with the old matrices embedded in the new one.

The bending displacement of the element relates to the nodal forces by the material properties of the beam as :

$$\begin{aligned} f_1 &= EI \frac{d^3 w(0)}{dx^3} & f_2 &= EI \frac{d^2 w(0)}{dx^2} \\ f_3 &= -EI \frac{d^3 w(0)}{dx^3} & f_4 &= EI \frac{d^2 w(0)}{dx^2} \end{aligned}$$

**Equation C- 8**

The element stiffness matrix is determined by substituting Equation C-6 into Equation C-8 and relating the force vector and displacement vector in the following fashion:

$$[k] = \begin{Bmatrix} w_1 \\ h\theta_1 \\ w_2 \\ h\theta_2 \end{Bmatrix}^{-1} \begin{Bmatrix} f_1 \\ f_2/h \\ f_3 \\ f_4/h \end{Bmatrix} = \frac{EI}{h^3} \begin{bmatrix} 12 & 6 & -12 & 6 \\ 6 & 4 & -6 & 2 \\ -12 & -6 & 12 & -6 \\ 6 & 2 & -6 & 4 \end{bmatrix}$$

**Equation C- 9**

The element stiffness matrix can also be derived from the expression for the potential energy for the element as:

$$[k] = \int_0^h EI \{L''(x)\} \{L''(x)\}^T dx$$

**Equation C- 10**

where  $L(x)$  is the vector of interpolating functions. Similarly, the element mass matrix is derived from the expression for the kinetic energy for the element as:

$$[m] = \int_0^h m \{L(x)\} \{L(x)\}^T dx = \frac{mh}{420} \begin{bmatrix} 156 & 22 & 54 & -13 \\ 22 & 4 & 13 & -3 \\ 54 & 13 & 156 & -22 \\ -13 & -3 & -22 & 4 \end{bmatrix}$$

**Equation C- 11**

Since the beam of interest is a rotating rotor blade, the equation of motion will require expressions for the element centrifugal stiffness, aerodynamic damping and applied forces. The rotating blade sees an increased stiffness due to centrifugal force, which increases the element strain energy by:

$$V_c = \frac{1}{2} \iiint \left\{ \sigma_x \left( \frac{\partial w}{\partial x} \right)^2 + 2\tau_{xy} \frac{\partial w}{\partial x} \frac{\partial w}{\partial y} \right\} dx dy dz$$

**Equation C- 13**

Under the assumption that the blade is a slender beam,  $\partial w / \partial y$  equals zero and the second term in **Equation C-12** is removed. The local stress  $\sigma_x$  is dependent only upon the local axial tension force  $T(x)$  such that the element centrifugal stiffness matrix takes the form:

$$[k_c] = \int_r^{r+h} T(x) \{L'(x)\}^T \{L'\} dx$$

**Equation C- 12**

where  $T(x)$  is given by:

$$T(x) = \int_x^h m \Omega^2 (x+h) dx + \int_{r+h}^L m \Omega^2 x dx$$

**Equation C- 14**

The combination of **Equations C-13** and **C-14** results in a different centrifugal stiffness matrix for each element based on its distance from the center of rotation.

Using a simplified version of quasi-steady strip theory, the aerodynamic load per unit span for a blade is given by:

$$L = -\frac{1}{2} \rho_\infty \frac{dC_l}{d\alpha} (\text{chord}) \Omega r \dot{w}$$

**Equation C- 15**

From Equation C-15, the expression for aerodynamic damping can be determined by substituting the interpolating polynomial vector as follows:

$$\{p\} = e^{i\omega t} \int_0^h p\{L(x)\} dx$$

**Equation C- 16**

Similarly, an applied harmonic load may be expressed as:

$$[c] = \frac{1}{2} \rho_{\infty} \frac{dC_l}{d\alpha} \Omega(\text{chord}) \int_0^h (r+h) \{L(x)\}^T \{L(x)\} dx$$

**Equation C- 17**

The global mass, stiffness, damping and forcing matrices describing the blade are developed by assembling the corresponding matrices from the individual elements in an overlapping pattern. The global matrices, denoted by capital letters, are used to develop an equation of motion for the entire blade based upon the motion of the individual nodes as follows:

$$[M]\{\ddot{w}\} + [C]\{\dot{w}\} + ([K_E] + [K_C])\{w\} = \{P\}$$

**Equation C- 18**

Since solution of the equation of motion will be effected using modal analysis, a transformation into a coordinate system of orthogonal modes is required. The matrices of eigenvectors and eigenvalues representing these orthogonal modes are obtained by matrix iteration of the following equation representing the eigenvalue problem for the undamped free vibration of the blade:

$$\lambda_i [u]_i = \frac{[M]}{([K_E] + [K_C])} [u]_i \quad \text{where} \quad \lambda_i = \frac{1}{\omega_{n_i}^2}$$

**Equation C- 19**

The equation of motion is uncoupled by a transformation into modal coordinates using the full matrix of orthogonal eigenvectors  $[U]$  according to:

$$\{q\} = [U]\{w\}$$

**Equation C- 20**

The  $[U]$  matrix is normalized such that

$$[U]^T [M] [U] = [I] \quad \text{and} \quad [U]^T ([K_E] + [K_C]) [U] = [\omega_n^2]$$

**Equation C- 22**

The equation of motion for the rotor blade is rewritten as:

$$[I]\{\ddot{q}\} + [C^*]\{\dot{q}\} + [\omega_n^2]\{q\} = \{P^*\}$$

**Equation C- 21**

where

$$[C^*] = [U]^T [C] [U] \quad \text{and} \quad \{P^*\} = [U]^T \{P\}$$

**Equation C- 24**

If the excitation of the rotor blade is harmonic in nature, i.e.

$$\{P^*\} = \{P_0^*\} e^{i\omega t}$$

**Equation C- 23**

then the assumption is made that the blade response will also be harmonic such that

$$\{q\} = \{q_0\} e^{i\omega t}$$

**Equation C- 25**

Taking derivatives of **Equation C-25** with respect to time, substituting the resulting expressions into **Equation C-22** and rearranging the result into a more convenient form yields:

$$([\omega_n^2 - \omega^2] + i\omega[C^*])\{q_0\} = \{P^*\}$$

**Equation C- 26**

The modal response  $\{q\}$  transforms back to nodal coordinates  $\{w\}$  using **Equation C-20**. The interpolating polynomials are applied to the resulting nodal responses to determine the displacement over the length of each element comprising the blade. The slope, shear and moment distribution over the length of the blade can be determined from the derivatives of the displacement equation (**Equation C-5**). (Rutkowski, 1983)



## APPENDIX D. FLOQUET THEORY

The second order ordinary differential equation describing the rigid flapping motion of a helicopter rotor blade can be considered linear, time-variant under certain conditions. This linear, time variant type of equation takes the general form:

$$\dot{\bar{x}} = A(t)\bar{x}$$

### Equation D-1

in state space where the coefficient  $A$  is a function of time such that  $A(t+T)=A(t)$  for a given period  $T$ . The analysis of such a periodic coefficient equation can be accomplished using several methods, one of which is Floquet theory. The solution of **Equation D-1** is of the form:

$$\bar{x}(t) = \phi(t, t_0)\bar{x}(t_0)$$

### Equation D-2

where  $\phi(t, t_0)$  is the Floquet state transition matrix relating coefficient values at the current time  $t$  to those at time  $t_0$ . Substituting **Equation D-2** into **Equation D-1** results in the following differential equation for  $\phi$ :

$$\dot{\phi}(t) = A(t)\phi(t)$$

### Equation D-3

which has initial conditions  $\phi(t_0, t_0)=[I]$ . Since  $\phi(t+T, t_0)$  must be a linear combination of  $\phi(t, t_0)$  in order to provide a solution to **Equation D-2**, then it follows that:

$$\phi(t+T, t_0) = \phi(t, t_0)\alpha \quad \text{and} \quad \phi(t+nT, t_0) = \phi(t, t_0)\alpha^n$$

### Equation D-4

where  $\alpha$  is some constant matrix dependent only upon the system. **Equation D-4** also implies that information about the solution for all time is contained in the state transition matrix for a single period. Floquet theory does not give the solution to the differential equation of motion but only information about properties of the solution. For example, information about the stability of the system is contained in the eigenvalues of the state transition matrix. In order to retrieve this information, re-write  $\phi(t, t_0)$  as:

$$\phi(t, t_0) = P(t)e^{\beta t} \quad \beta \equiv \frac{1}{T} \ln \alpha$$

### Equation D-5

where  $P(t)$  is a purely periodic matrix and  $\beta$  represents exponential growth or decay of the solution and thus the stability of the system over time. Therefore, the eigenvalues of  $\alpha$  relate to the eigenvalues of  $\beta$  by the equation:

$$\Lambda = \frac{1}{T} \ln \Theta$$

**Equation D-6**

where  $\Lambda$  is the matrix of eigenvalues of  $\beta$  and  $\Theta$  is the matrix of eigenvalues of  $\alpha$ . In practice, **Equation D-3** is numerically integrated over the period  $t=0$  to  $t=T$  from the initial conditions of  $\phi(t_0, t_0)$  equal to the identity matrix  $I$ . The eigenvalues of the resulting  $\alpha$  matrix are determined and converted into roots of the system according to **Equation D-6**. The system roots  $\Lambda$  contain the frequency and damping information needed to determine system stability. The real components of  $\Lambda$  represent system damping and must be less than zero or the system is unstable. The imaginary components of  $\Lambda$  represent the natural frequencies of the system which, according to Floquet theory, cannot be determined uniquely but may be determined harmonically as:

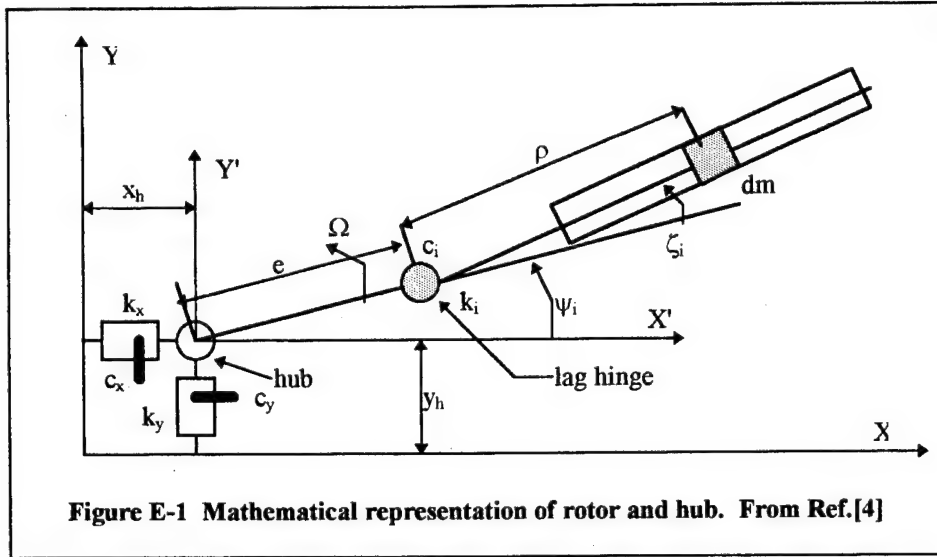
$$\omega_i = \text{imaginary}(\lambda_i) + n \frac{2\pi}{T}$$

**Equation D-7**

To uniquely determine the natural frequency requires knowledge of the behavior of a particular mode. (Johnson, 1980)

## APPENDIX E. BLADE LAG EQUATIONS OF MOTION

The equations describing the rigid body lag motion of rotor blades attached to a flexible hub have coefficients that are periodic in nature. Two common methods of dealing with periodic coefficients are first, to apply Floquet theory as in the case of rotor flapping stability or second, to make the assumptions and transformations necessary to remove the periodicity. The first method has the drawback of adding complexity to the analysis. The second method results in a more simplistic, and thus limited, model. For preliminary design and the qualitative study of rotor system lag behavior, the constant coefficient method is sufficient. The rotor blades and hub are mathematically represented by the arrangement in **Figure E-1**.



In developing the equations of motion for an  $n$  bladed rotor, the first mass moment of inertia  $S_b$  and the second mass moment of inertia  $I_b$  of the rotor blade are defined as:

$$S_b = \int \rho dm \quad I_b = \int \rho^2 dm$$

**Equation E-1**

Introducing the following parameters to simplify the blade equation notation:

$$v_0^2 = e \frac{S_b}{I_b} \quad \omega_0^2 = \frac{k_i}{I_b} \quad \eta_i = \frac{c_i}{I_b}$$

**Equation E-2**

The linearized blade equations incorporating **Equations E-1** and **E-2** and assuming small displacements are:

$$\ddot{\zeta}_i + \eta_i \dot{\zeta}_i + (\omega_{0i}^2 + \Omega^2 v_0^2) \zeta_i = \left( \frac{v_0^2}{e} \right) [\ddot{x}_h \sin \psi_i - \ddot{y}_h \cos \psi_i] \quad i = 1, 2, \dots, n$$

**Equation E-3**

where the subscript  $i$  represents the  $i$ th blade and the subscript  $h$  represents the hub motion as described in **Figure E-1**. The hub is subjected to forces due to accelerations of the rotor center of mass in the  $x$ - and  $y$ -directions according to:

$$\begin{aligned} m_x \ddot{x}_h + c_x \dot{x}_h + k_x x_h &= -nm_b \ddot{x}_c \\ m_y \ddot{y}_h + c_y \dot{y}_h + k_y y_h &= -nm_b \ddot{y}_c \end{aligned}$$

**Equation E-4**

where the center of mass components are denoted by the subscript  $c$  and the hub is assumed uncoupled in the absence of the rotor. The individual blade centers of mass are located at:

$$\begin{aligned} x_{i_c} &= e \cos \psi_i + \rho_c \cos(\psi_i + \zeta_i) \\ y_{i_c} &= e \sin \psi_i + \rho_c \sin(\psi_i + \zeta_i) \end{aligned}$$

**Equation E-6**

and the rotor center of mass has coordinates:

$$\begin{aligned} x_c &= x_h - \left( \frac{\rho_c}{n} \right) \sum_{i=1}^n \zeta_i \sin \psi_i \\ y_c &= y_h + \left( \frac{\rho_c}{n} \right) \sum_{i=1}^n \zeta_i \cos \psi_i \end{aligned}$$

**Equation E-5**

At this point the hub equations are in a fixed coordinate system and the blade equations are in a rotating coordinate system. In order to make use of a constant coefficient approximation for the combined hub and rotor system, the hub equations must be transformed into the rotating frame to remove the periodic nature of the coefficients. This transformation requires the assumption that the hub is isotropic in nature, ie.  $m_x = m_y$ ,  $c_x = c_y$  and  $k_x = k_y$ . The hub equations are transformed into the rotating system according to:

$$\begin{aligned}\bar{x} &= x_h \cos \Omega t + y_h \sin \Omega t \\ \bar{y} &= -x_h \sin \Omega t + y_h \cos \Omega t\end{aligned}$$

**Equation E-8**

The following relations are developed by differentiating **Equation E-7** with respect to time:

$$\begin{aligned}\dot{x}_h \cos \Omega t + \dot{y}_h \sin \Omega t &= \dot{\bar{x}} - \Omega \bar{y} \\ -\dot{x}_h \sin \Omega t + \dot{y}_h \cos \Omega t &= \dot{\bar{y}} + \Omega \bar{x} \\ \ddot{x}_h \cos \Omega t + \ddot{y}_h \sin \Omega t &= \ddot{\bar{x}} - \Omega^2 \bar{x} - 2\Omega \dot{\bar{y}} \\ -\ddot{x}_h \sin \Omega t + \ddot{y}_h \cos \Omega t &= \ddot{\bar{y}} - \Omega^2 \bar{y} + 2\Omega \dot{\bar{x}}\end{aligned}$$

**Equation E-7**

Differentiating **Equation E-6** twice with respect to time, applying the relations of **Equation E-8** and substituting the result into **Equation E-4** results in the following expression for hub motion in the rotating frame:

$$\begin{aligned}\ddot{\bar{x}} + \eta_h \dot{\bar{x}} + (\omega_h^2 - \Omega^2) \bar{x} - 2\Omega \eta_h \dot{\bar{y}} - \Omega \eta_h \bar{y} &= v_h^2 \sum_{i=1}^n \left[ \left( \ddot{\zeta}_i - \Omega^2 \zeta_i \right) \sin \frac{2\pi}{n} (i-1) + 2\Omega \dot{\zeta}_i \cos \frac{2\pi}{n} (i-1) \right] \\ \ddot{\bar{y}} + \eta_h \dot{\bar{y}} + (\omega_h^2 - \Omega^2) \bar{y} + 2\Omega \eta_h \dot{\bar{x}} + \Omega \eta_h \bar{x} &= -v_h^2 \sum_{i=1}^n \left[ \left( \ddot{\zeta}_i - \Omega^2 \zeta_i \right) \cos \frac{2\pi}{n} (i-1) - 2\Omega \dot{\zeta}_i \sin \frac{2\pi}{n} (i-1) \right]\end{aligned}$$

**Equation E-9**

**Equation E-9** makes use of the following parameters for simplification:

$$v_h^2 = \frac{S_b}{m_x + nm_b} \quad \omega_h^2 = \frac{k_x}{m_x + nm_b} \quad \eta_h = \frac{c_x}{m_x + nm_b}$$

**Equation E-10**

The relations of **Equation E-8** are also applied to **Equation E-3**, the blade equations, resulting in the following expression:

$$\ddot{\zeta}_i + \eta_i \dot{\zeta}_i + (\omega_{oi}^2 + \Omega^2 v_{o2}) \zeta_i = \frac{v_o^2}{e} \left[ \left( \ddot{\bar{x}} - \Omega^2 \bar{x} - 2\Omega \dot{\bar{y}} \right) \sin \frac{2\pi}{n} (i-1) - \left( \ddot{\bar{y}} - \Omega^2 \bar{y} + 2\Omega \dot{\bar{x}} \right) \cos \frac{2\pi}{n} (i-1) \right]$$

**Equation E-11**

for  $i=1, 2, \dots, n$  rotor blades. **Equations E-9 and E-11** are in their final form, with all periodic dependence removed. The resulting  $n+2$  equations are the constant coefficient approximation to the rotor blade lag motion. Hammond (1974)

## **APPENDIX F. COMPUTER PROGRAMS**

## A. DYNAM.M

```
% DYNAM.M
% JANRAD: NPS Helicopter Preliminary Design Program
% Rotor Blade Dynamic Response Routines
% Written by Lt Juan D. Cuesta, revised by LT DAN HIATT
% September 1994, revised February 1995
% This program was designed as an interactive preliminary design tool for rotor blade dynamic analysis
and design of either an articulated or hingeless rotor blade system. The program provides the shear,
moment, slope angle, and deflection of the flatwise response at any point along the length of a rotor blade
to the steady and first ten harmonic aerodynamic loads. This data can then be plotted at various azimuth
blade positions. The input of variables follows the same format as written by Majors Bob Nicholson, Jr. and
Walter Wirth, Jr. for JANRAD.
```

```
% Variable List for Dynam.m, Blade.m, output.m
```

```
% a      lift curve slope
% alphaFn elemental force column matrix
% az      azimuth position angle
% bc      root boundary condition
% cblade2 blade chord at radial segment, from tip to root
% Cn      flatwise aerodynamic damping on blade element
% delr    blade radial segment width, starting from blade hinge
% dfn     imaginary component of harmonic thrust terms
% dFn     real component of harmonic thrust terms
% dFo     steady state thrust terms
% dT      JANRAD Performance routine thrust output
% EI      elemental bending modulus
% Exx,Eyy elemental modulus of elasticity
% filename1 .mat file with janrad data
% filename3 .mat file which contains blade data
% Fn      distribution of thrust airloads from tip to root
% Ixx,Iyy distribution of moment of inertia of blade elements
% lsn     length of blade segment
% mn      distribution of blade mass
% Mn      flatwise moment for blade element
% naz     number of azimuth sectors
% nbe     number of blade elements
% omega   rotor rotational velocity
% omegae  excitation frequency
% Pmat    running transfer matrix along blade length
% psi     azimuth angle
% rho     ambient air density
% rn      radius, rotor blade radial segment, from blade hinge
% Sn      flatwise shear for blade element
% Thetan  flatwise slope of blade element
% tip_art_bc articulated blade tip slope and deflection bound. cond.
% tip_rig_bd hingeless blade tip slope and deflection bound. cond.
% Tn      elemental radial tension
```



```

% Un    Transfer matrix between adjacent blade elements
% view  option variable for viewing choice
% Wn    distribution of blade weight
% X,Y,Yout  output data to generate graphs
% Yn    flatwise deflection of blade element
% Zn    rotor blade element state vector
% Zroot  rotor blade root state vector
% Ztip   rotor blade tip state vector

flag=1;
flag=exist('filename1');
if flag == 0,
    disp(' ')
    disp(' *** You must run Rotor Performance Analysis first ***')
    disp(' *** Press Any Key to Continue ***')
    disp(' ')
    pause
else,
    eval(['load ',filename1 '_p'])
    eval(['load ',filename1])
    acdat=filename1
    disp(' *** ROTOR BLADE DYNAMIC ANALYSIS ROUTINE ***')
    disp(' Do you want to edit an existing blade file or create a new one?')
    answer3=input(' 1. edit existing file  2. create new file  >>');
    if answer3==1,
        clc
        disp(' *** LOAD INSTRUCTIONS *** ')
        disp(' A. Input the name of the rotor blade data file to edit.')
        disp(' B. The file was saved in your previous session')
        disp(' with a ".mat" extension.')
        disp(' C. Do not include the extension or quotations.')
        disp(' ex: blade1')
        flag=0;
        while flag < 1
            filename3=input(' Enter the name of Blade data input file: ','s');
            blddat=filename3;
            eval(['flag=exist("'",filename3,'.mat"');'])
            if flag < 1,
                disp(' The file does not exist, try again or < Ctrl-C >')
                disp(' to exit program.')
            end
        end
        eval(['load ',filename3])
        check4=1;
        while check4 > 0,
            clc
            disp(' ***BLADE DYNAMICS EDIT MENU ***')
            disp(' 1. root boundary condition')
            disp(' 2. blade material properties')
            disp(' 3. weight distribution')
            disp(' 4. blade chord')

```

```

disp('      *** ALL OTHER BLADE INFORMATION IS ENTERED IN MAIN JANRAD MENU ***')
disp('      0. NO CHANGES')
choice1=input('      Input the parameter to change: ');
if isempty(choice1),
    choice1=0;
end
if choice1==1,
    clc
    disp('Root Boundary Condition')
    bc
    tmp=bc;
    flag=1;
    while flag > 0
        bc=input('Root Boundary Condition 1. Articulated 2. Hingeless >> ');
        if isempty(bc),
            bc=tmp;
        end
        if bc==1,
            flag=0;
        elseif bc==2,
            flag=0;
        else
            disp(' *** Enter a 1 or 2 ***')
        end
    end
end
if choice1==2,
    clc
    disp('1. variable elasticity, E, and variable moment of inertia, I')
    disp('2. variable stiffness, EI')
    option=input('Choose 1. OR 2. >> ');
    clc
    disp('1. Flatwise component, Ey Iyy or Ely')
    disp('2. Edgewise component, Ex Ixx or Elx')
    suboption=input('Choose 1. OR 2. >> ');
    if option==1,
        if suboption==1
            En=Ey;
        else
            En=Ex;
        end
    end
    clc
    E=En/1e6
    tmp=En;
    disp('1) Enter as a row vector starting from the')
    disp(' tip and ending with the root; ex: "[18 18.1 .... 21]"')
    fprintf('2) YOU MUST ENTER%3.0f ELEMENTS, <CR> IF NO CHANGES ARE MADE\n',nbe)
    disp('3) Enter the modulus of ELASTICITY, E, distribution (lbs/in^2 x 10^6): ')
    En=input(' >>').*1e6;
    if isempty(En),
        En=tmp;
    end
end

```

```

end
while (length(En)~=nbe),
    disp('1) Enter as a row vector starting from the ')
    disp(' tip and ending with the root; ex: "[18 18.1 .... 21]"')
    fprintf('2) YOU MUST ENTER%3.0f ELEMENTS, <CR> IF NO CHANGES ARE MADE\n',nbe)
    disp(' ')
    disp('3) Enter the modulus of ELASTICITY, E, distribution (lbs/in^2 x 10^6): ')
    En=input(' >>').*1e6;
end
if suboption==1
    Ey=En;
else
    Ex=En;
end
clc
if suboption==1
    Ibn=Iyy;
else
    Ibn=Ixx;
end
Ibn
tmp=Ibn;
disp('1) Enter as a row vector starting from the ')
disp(' tip and ending with the root; ex: "[3.9 4.09 .... 15.1]"')
disp(' ')
fprintf('2) YOU MUST ENTER%3.0f ELEMENTS, <CR> IF NO CHANGES ARE MADE\n',nbe)
disp(' ')
disp('3) Enter blade moment of INERTIA, Ixx or Iyy, distribution (in^4): ')
Ibn=input(' >>');
if isempty(Ibn),
    Ibn=tmp;
end
while (length(Ibn)~=nbe),
    disp('1) Enter as a row vector starting from the ')
    disp(' tip and ending with the root; ex: "[3.9 4.09 .... 15.1]"')
    disp(' ')
    fprintf('2) YOU MUST ENTER%3.0f ELEMENTS, <CR> IF NO CHANGES ARE MADE\n',nbe)
    disp(' ')
    disp('3) Enter blade moment of INERTIA, Ixx or Iyy, distribution (in^4): ')
    Ibn=input(' >>');
end
if suboption==1
    Iyy=Ibn;
else
    Ixx=Ibn;
end
if exist('Elx')==1
    clear Elx Ely
end
if option==2,

```

```

clc
if suboption==1
    EI=EIy;
else
    EI=EIx;
end
EI
tmp=EI
disp('1) Enter as a row vector starting from the ')
disp(' tip and ending with the root; ex: "[70.2 73.62 .... 271.8]"')
disp(' ')
fprintf('2) YOU MUST ENTER%3.0f ELEMENTS, <CR> IF NO CHANGES ARE MADE\n',nbe)
disp(' ')
disp('3) Enter the bending/stiffness modulus, EI, distribution (lbs in^2 x 10^6): ')
EI=input(' >>').*1e6;
if isempty(EI),
    EI=tmp;
end
while (length(EI)~=nbe),
    disp('1) Enter as a row vector starting from the ')
    disp(' tip and ending with the root; ex: "[70.2 73.62 .... 271.8]"')
    disp(' ')
    fprintf('2) YOU MUST ENTER%3.0f ELEMENTS, <CR> IF NO CHANGES ARE MADE\n',nbe)
    disp(' ')
    disp('3) Enter the bending stiffness, EI, distribution (lbs in^2 x 10^6): ')
    EI=input(' >>').*1e6;
end
if suboption==1
    EIy=EI;
else
    EIx=EI;
end
if exist('Ex')
    clear Ex Ey Ixx Iyy
end
end
end
if choice1==3,
    clc
    Wn
    tmp=Wn;
    disp(' ')
    fprintf('1) YOU MUST ENTER%3.0f ELEMENTS, <CR> IF NO CHANGES ARE MADE\n',nbe)
    disp(' ')
    disp('2) Enter as a row vector starting from the ')
    disp(' tip and ending with the root; ex: "[9.86 9.95 .... 11.96]"')
    disp(' ')
    disp('3) THE TOTAL WEIGHT MUST BE GREATER THAN THE AERODYNAMIC')
    fprintf(' PORTION OF THE BLADE: %6.2f\n',wblade)
    disp(' ')
    disp('weight distribution (lbs/in): ')

```

```

Wn=input(' >>');
if isempty(Wn),
    Wn=tmp;
end
while (length(Wn)~=nbe),
    disp(' ')
    fprintf('1) YOU MUST ENTER%3.0f ELEMENTS, <CR> IF NO CHANGES ARE MADE\n',nbe)
    disp(' ')
    disp('2) Enter as a row vector starting from the ')
    disp(' tip and ending with the root; ex: "[9.86 9.95 .... 11.96]"')
    disp(' ')
    disp('3) THE TOTAL WEIGHT MUST BE GREATER THAN THE AERODYNAMIC')
    fprintf(' PORTION OF THE BLADE: %6.2f\n',wblade)
    disp(' ')
    disp('weight distribution (lbs/in): ')
    Wn=input(' >>');
end
end
if choice1==4,
    clc
    if exist('cblade')
        cblade
        tmp=cblade;
    end
    disp(' ')
    fprintf('1) YOU MUST ENTER%3.0f ELEMENTS, <CR> IF NO CHANGES ARE MADE\n',nbe)
    disp(' ')
    disp('2) Enter as a row vector starting from the ')
    disp(' tip and ending with the root; ex: "[1.5 1.95 .... 0.96]"')
    disp(' ')
    disp('blade chord (ft): ')
    cblade=input(' >>');
    if isempty(cblade),
        cblade=tmp;
    end
    while (length(cblade)~=nbe),
        disp(' ')
        fprintf('1) YOU MUST ENTER%3.0f ELEMENTS, <CR> IF NO CHANGES ARE MADE\n',nbe)
        disp(' ')
        disp('2) Enter as a row vector starting from the ')
        disp(' tip and ending with the root; ex: "[1.5 1.95 .... 0.96]"')
        disp(' ')
        disp('blade chord (ft): ')
        cblade=input(' >>');
    end
end
if choice1==0,
    clc
    disp(' ')
    disp(' ')
    disp(' *** SAVE INSTRUCTIONS ***')

```

```

disp(' ')
disp('    A. Save the new data to a specified file name.')
disp('    B. Do not use an extension or quotations.')
disp('    C. Use letter/number combinations of 6 characters or less.')
disp('    D. The file will be saved with a ".mat" extension.')
disp(' ')
disp('    ex: blade1')
disp(' ')
disp('    E. If you made no changes, press < Enter >, the file will')
disp('        be saved with the original name.')
    flag=1;
    while flag > 0
        filename0=filename3;
        filename3=input('        save file as: ','s');
        if isempty(filename3),
            filename3=filename0;
        end
        clear filename0
        if length(filename1) > 6,
            disp(' ')
            disp('        use 6 characters or less')
            flag=1;
        else
            flag=0;
        end
        eval(['save ',filename3,' bc Ex Ixx Ey Iyy Wn cblade'])
        check4=0;
    end
end
end
end

% Creating a new file

if answer3==2,
    change=1;
    while change > 0,
        clc
        bc=input('Root Boundary Condition 1. Articulated 2. Hingeless >> ');
        while isempty(bc),
            disp(' ')
            disp('You must enter a numerical value')
            bc=input('Root Boundary Condition 1. Articulated 2. Hingeless >> ');
        end
        disp(' ')
        disp('    Do you want to enter:')
        disp(' ')
        disp('    1. elasticity, E, AND moment distribution, Ixx/Iyy')
        disp('    OR')
        disp('    2. just the bending stiffness, EI, distribution.')
        clear option1

```

```

option1=input('Enter 1 or 2 >> ');
while isempty(option1),
    option1=input('Enter 1 or 2 >> ');
end
clc
for i=1:2
    if i==1
        disp(' ')
        disp('You will be entering the flatwise components of modulus,')
        disp(' Ey, and moment of inertia, Iyy, or EIy on this pass')
    else
        disp(' ')
        disp('You will be entering the edgewise components of modulus,')
        disp(' Ex, and moment of inertia, Ixx, or Elx on this pass')
    end
    if option1==1,
        disp(' ')
        disp('1) Enter as a row vector starting from the ')
        disp(' tip and ending with the root; ex: "[18 18.1 .... 21]"')
        disp(' ')
        fprintf('2) YOU MUST ENTER%3.0f ELEMENTS\n',nbe)
        disp(' ')
        disp('3) Enter the modulus of ELASTICITY, E, distribution (lbs/in^2 x 10^6):')
        En=input('>> ').*1e6;
        while (length(En)~=nbe),
            disp(' ')
            fprintf('1) YOU MUST ENTER%3.0f ELEMENTS\n',nbe)
            disp(' ')
            disp('2) Enter the modulus of ELASTICITY, E, distribution (lbs/in^2 x 10^6):')
            En=input('>> ').*1e6;
        end
        if i==1
            Ey=En;
        else
            Ex=En;
        end
        disp(' ')
        disp('1) Enter as a row vector starting from the ')
        disp(' tip and ending with the root; ex: "[3.9 4.09 .... 15.1]"')
        disp(' ')
        fprintf('2) YOU MUST ENTER%3.0f ELEMENTS\n',nbe)
        disp(' ')
        disp('3) Enter blade moment of INERTIA, Ixx/Iyy, distribution (in^4): ')
        Ibn=input('>> ');
        while (length(Ibn)~=nbe),
            disp(' ')
            fprintf('1) YOU MUST ENTER%3.0f ELEMENTS\n',nbe)
            disp(' ')
            disp('2) Enter blade moment of INERTIA, Ixx/Iyy, distribution (in^4): ')
            Ibn=input('>> ');
        end
    end
end

```

```

if i==1
    Iyy=Ibn;
else
    Ixx=Ibn;
end
end %(endif option1==1)
if option1==2,
    disp(' ')
    disp('1) Enter as a row vector starting from the ')
    disp(' tip and ending with the root; ex: "[70.2 73.62 .... 271.8]"')
    disp(' ')
    fprintf('2) YOU MUST ENTER%3.0f ELEMENTS\n',nbe)
    disp(' ')
    disp('3) Enter the bending stiffness, EI, distribution (lbs in^2 x 10^6): ')
    EI=input('>> ').*1e6;
    while (length(EI)~=nbe),
        disp(' ')
        fprintf('1) YOU MUST ENTER%3.0f ELEMENTS\n',nbe)
        disp('2) Enter the bending stiffness, EI, distribution (lbs in^2 x 10^6): ')
        EI=input('>> ').*1e6;
    end
end %(endif option1==2)
if i==1
    EIy=EI;
else
    Elx=EI;
end
end %(end for i=1:2)
disp(' ')
disp('1) Enter as a row vector starting from the ')
disp(' tip and ending with the root; ex: "[9.86 9.95 .... 11.96]"')
disp(' ')
fprintf('2) YOU MUST ENTER%3.0f ELEMENTS\n',nbe)
disp(' ')
disp('3) THE TOTAL WEIGHT MUST BE GREATER THAN THE AERODYNAMIC')
fprintf(' PORTION OF THE BLADE: %6.2f\n',wblade)
disp(' ')
disp('Enter weight distribution (lbs/in)')
Wn=input('>> ');
while (length(Wn)~=nbe),
    disp(' ')
    fprintf('1) YOU MUST ENTER%3.0f ELEMENTS\n',nbe)
    disp('2) Enter weight distribution (lbs/in)')
    Wn=input('>> ');
end
clc
disp(' ')
fprintf('1) YOU MUST ENTER%3.0f ELEMENTS, PRESS <ENTER> IF NO CHANGES ARE
MADE\n',nbe)
disp(' ')
disp('2) Enter as a row vector starting from the ')

```



```

disp(' tip and ending with the root; ex: "[1.5 1.95 .... 0.96]"')
disp(' ')
disp('blade chord (ft): ')
cblade=input(' >>');
while (length(cblade)~=nbe),
    disp(' ')
    fprintf('1) YOU MUST ENTER%3.0f ELEMENTS, PRESS <ENTER> IF NO CHANGES ARE
MADE\n',nbe)
    disp(' ')
    disp('2) Enter as a row vector starting from the ')
    disp(' tip and ending with the root; ex: "[1.5 1.95 .... 0.96]"')
    disp(' ')
    disp('blade chord (ft): ')
    cblade=input(' >>');
end
clc
disp(' ')
disp(' ')
disp(' *** DATA ENTRY COMPLETE ***)
disp(' PLEASE REVIEW YOUR DATA')
disp(' ')
disp(' PRESS ANY KEY TO CONTINUE')
pause
if option1==1,
    Ex=Ex/1e6
    Ixx
    Ey=Ey/1e6
    Iyy
end
if option1==2,
    Elx
    Ely
end
Wn
cblade
disp('Do you wish to make any changes?')
change=input('0. No 1. Yes >>');
end

clc
disp(' ')
disp(' ')
disp(' *** SAVE INSTRUCTIONS ***)
disp(' ')
disp(' A. Save the data to a specified file name.')
disp(' B. Do not use an extension or quotations.')
disp(' C. Use letter/number combinations of 6 characters or less.')
disp(' D. The file will be saved with a ".mat" extension.')
disp(' ')
disp(' ex: blade1')
disp(' ')

```

```

disp('      E. If you do not enter a name, the default is "blade1" ')
flag=1;
while flag > 0
    filename3=input('      save file as: ','s');
    if isempty(filename3),
        filename3='blade1';
    end
    if length(filename3) > 6,
        disp(' ')
        disp('      use 6 characters or less')
        flag=1;
    else
        flag=0;
    end
end
if exist('Elx')==0,
    eval(['save ',filename3,' bc Ex Ixx Ey Iyy Wn cblade'])
end
if exist('Ex')==0,
    eval(['save ',filename3,' bc Elx Elx Wn cblade'])
end
blddat=filename3;
end
clc
disp(' ')
disp(' ')
disp('Do you wish to make a fan plot of the rotor blade natural frequencies?')
disp(' ')
disp('      enter 1 for no or 2 for yes')
disp(' ')
choice=input(' >>')
if choice==2
    mykbis
else
    bldrev
end
end
end

```

## B. MYKBIS.M

% This sub-program determines the natural frequencies of the rotor blade over a range of rotational speed from 10 rad/sec to 1.5\*operating speed. The method used is a step followed by bisection after the root interval is determined.

% written by LT DAN HIATT, FEBRUARY 1995

```

clc
%disp('This program will develop the fan plot of natural frequencies')
%disp('for the rotor blade. The process may take some time.')
%disp(' ')

```

```

eval(['load ',acdat ])
eval(['load ',[acdat '_p']])
eval(['load ',blddat])
clear resid freq indx1 indx2

global Ex Ey Ixx Iyy Wn R thetao twist nbe e cblade a Cthta bc
global Vxx vzz uzz gzz Uxx Gxx Uxz Vxz Gxz vxz uzx gzx

omega0=omega;
kk=1;
for ii=10:3:round(1.5*omega)
    indx1(kk)=ii/omega0;
    omega=ii
    indx2=0;
    resid(kk,1)=bldfan(omega,1);
    jj=1;
    for k=1:5:150
        jj=jj+1;
        omegae=k;
        resid(kk,jj)=bldfan(omega,omegae);

        if sign(resid(kk,jj-1))~=sign(resid(kk,jj))
            l0=k-5;
            r0=k;
            indx2=indx2+1;

% Initialize the midpoint outside of the interval and
% evaluate the polynomial at the endpoints.

            xm =r0+1;
            y1 = bldfan(omega,l0);
            y3 = bldfan(omega,r0);

% Test for opposing signs for the function evaluated at
% the end points to ensure interval contains a root.

            if sign(y1)==sign(y3)
                error('The interval given does not bracket a real root');
            end

% Set xm1 to previous midpoint for convergence test
% and find the new interval midpoint.

            for i=1:100
                xm1=xm;
                xm=(l0+r0)*0.5;
                y1=bldfan(omega,l0);
                y2=bldfan(omega,xm);
                if abs(xm1-xm) > .2 % convergence test
                    if sign(y1)==sign(y2) % test for root in left interval
                        l0=xm; % reset interval to left or...
                    end
                end
            end
        end
    end
end

```

```

        else
            r0=xm;          % reset interval to right
        end
    else
        b=xm;              % set output to current midpoint
        break
    end % (end if)
end % (end for i=1:100)
freq(kk,indx2)=b/omega0;
if i == 100
    error('No root found in 100 steps for the interval given.');
```

end

% Output results

```

    end % (end if sign~=sign)
    end % (end for k=10:5:100)
    kk=kk+1;
end % (end for ii=10:5:?)
freq
indx1
plot(indx1,freq(:,1:4))
%grid on
hold on
for i=1:3
    plot(indx1,indx1.*i,')
end
title('FAN PLOT FOR ROTOR BLADE NATURAL FREQUENCIES')
xlabel('Rotational Speed Ratio - omega/omega_0')
ylabel('Natural Frequency Ratio - omega_n/omega_0')
return
```

### C. BLDFAN.M

```
function [resid]=bldfan(omega, omegae)
```

```
% bldfan(omega,omegae)
```

```
%
```

% Myklestad-Thomson based blade dynamic response program which incorporates articulated and cantilever root conditions. Also included are lag damping and coupled flap and lag responses due to twist. Not included are torsional responses. This function develops the fan plot of natural frequencies for the blade.

```
% LT DAN HIATT, MARCH 1995
```

```
rho=.0023778;
```

```
global Ex Ey Ixx Iyy Wn R thetao twist nbe e cblade a Cthta bc
```

```
global Vxx vzz uzz gzz Uxx Gxx Uxz Vxz Gxz vxz uzx gzx
```

```
% enter lag damping information if needed
```

```
if exist('Cthta')==0
```

```
    disp('Enter the value for lag damper damping coefficient (lb-ft/sec)')
```

```
    disp('enter zero if no lag damper is installed')
```

```

Cthta=input(' >>');
end

if exist('Ex')==1
    Elx=Ex.*Ixx;
    Ely=Ey.*Iyy;
end

thetao=thetao*pi/180; % collective at .7*R
delr=(R-e)/nbe;
deltwst=twist/nbe; % change in blade angle due to twist per element
Beta0=thetao+twist/(R-e)*.3*R;

for k=nbe:-1:1,
    rn((nbe+1)-k)=e+k*delr-delr/2; % find radius at midpoint of each element
end

lsn=(rn(3)-rn(4))*12; % all element lengths are the same
mn=Wn*lsn/(32.174); % concentrated mass of an element (slug)

% Calculating the flatwise aerodynamic damping, Cn, the tension, Tn,
% and the combined collective and twist, Beta, for each element
cblade2=cblade;
Cn=zeros(1,nbe);
Tn=zeros(1,nbe);
Tn(1)=mn(1)*omega^2*rn(1); % calculating tension since it is independent
Beta(1)=Beta0; % collective setting and twist combined value at rotor tip

for k=1:nbe-1,
    Cn(k)=a*cblade2(k+1)*(rn(k)-rn(k+1)).*5*rho*omega*rn(k);
    Tn(k+1)=Tn(k) + mn(k+1)*omega^2*rn(k+1); % element tension vector
    Beta(k+1)=Beta0-k*deltwst; % collective and twist combined at element
end

% this section develops the twist coupling terms
if exist('Vxx')==0
    for k=1:nbe
        vn(k)=lsn/Ely(k); % flatwise slope due to moment
        un(k)=.5*lsn^2/Ely(k); % flatwise slope due to load
        gn(k)=lsn^3/(3*Ely(k)); % flatwise deflection due to load
        Vn(k)=lsn/Elx(k); % chordwise slope due to moment
        Un(k)=.5*lsn^2/Elx(k); % chordwise slope due to load
        Gn(k)=lsn^3/(3*Elx(k)); % chordwise deflection due to load
        vzz(k)=Vn(k)*(sin(Beta(k)))^2+vn(k)*(cos(Beta(k)))^2;
        uzz(k)=Un(k)*(sin(Beta(k)))^2+un(k)*(cos(Beta(k)))^2;
        gzz(k)=Gn(k)*(sin(Beta(k)))^2+gn(k)*(cos(Beta(k)))^2;
        Vxx(k)=vn(k)*(sin(Beta(k)))^2+Vn(k)*(cos(Beta(k)))^2;
        Uxx(k)=un(k)*(sin(Beta(k)))^2+Un(k)*(cos(Beta(k)))^2;
        Gxx(k)=gn(k)*(sin(Beta(k)))^2+Gn(k)*(cos(Beta(k)))^2;
        Vxz(k)=(Vn(k)-vn(k))*sin(Beta(k))*cos(Beta(k));
    end
end

```

```

    Uxz(k)=(Un(k)-un(k))*sin(Beta(k))*cos(Beta(k));
    Gxz(k)=(Gn(k)-gn(k))*sin(Beta(k))*cos(Beta(k));
    vzx(k)=Vxz(k); uzx(k)=Uxz(k); gzx(k)=Gxz(k);
end
end

j=sqrt(-1);

for kk=1:4 % pass thru the eqns 4 times to resolve the 4 root unknowns
    % thetai, thetaf, Z and X in terms of the tip values
    thtprvf=0;
    thtprve=0;
    Tprv=0;
    Zprv=0;
    Xprv=0;
    Sprvf=0; % boundary condition
    Sprve=0; % boundary condition
    Mprvf=0; % boundary condition
    Mprve=0; % boundary condition

    if kk==1
        thtprvf=1; % solving for theta flatwise coeff
    elseif kk==2
        Zprv=1; % solving for flatwise deflection coeff
    elseif kk==3
        thtprve=1; % solving for edgewise theta coeff
    elseif kk==4
        Xprv=1; % solving for edgewise deflection coeff
    end % (endif kk)

    for k=1:nbe,
        thtfl(k)=thtprvf*(1+Tprv*uzz(k))+Tprv*thtprve*uzx(k)-Mprvf*vzz(k)...
            -Mprve*vzx(k)-Sprvf*uzz(k)-Sprve*uzx(k);
        thtel(k)=thtprve*(1+Tprv*Uxx(k))+Tprv*thtprvf*Uxz(k)-Mprve*Vxx(k)...
            -Mprvf*Vxz(k)-Sprve*Uxx(k)-Sprvf*Uxz(k);
        Z1(k)=Zprv-thtfl(k)*lsn+Tprv*(thtprvf*gzz(k)+thtprve*gzx(k))...
            -Mprvf*uzz(k)-Mprve*uzx(k)-Sprvf*gzz(k)-Sprve*gzx(k);
        X1(k)=Xprv-thtel(k)*lsn+Tprv*(thtprve*Gxx(k)+thtprvf*Gxz(k))...
            -Mprve*Uxx(k)-Mprvf*Uxz(k)-Sprve*Gxx(k)-Sprvf*Gxz(k);
        Mfl(k)=Mprvf+Sprvf*lsn-Tprv*(Zprv-Z1(k));
        Mel(k)=Mprve+Sprve*lsn-Tprv*(Xprv-X1(k));

        % this section determines the shear component
        Sfl(k)=Sprvf+(mn(k)*omegae^2)*Z1(k)/12;
        Sel(k)=Sprve+mn(k)*(omegae^2+omega^2)*X1(k)/12;

        % this section renames the current values for looping
        thtprvf=thtfl(k);
        thtprve=thtel(k);
        Tprv=Tn(k);
        Zprv=Z1(k);
    end
end

```

```

Xprv=X1(k);
Sprvf=Sf1(k);
Sprve=Se1(k);
Mprvf=Mf1(k);
Mprve=Me1(k);

end % (end for k=1:nbe)

if bc==1 % assemble the soln matrix for articulated blades using root b.c.
    Pmat(1:4,kk)=[Mf1(nbe);Z1(nbe);Me1(nbe)-j*Cthta*omegae;X1(nbe)];
else % assemble the soln matrix for cantilever blades using root b.c.
    Pmat(1:4,kk)=[thtf1(nbe);Z1(nbe);thte1(nbe);X1(nbe)];
end
end % (end for kk=1:4)

% this section determines the residual
resid=det(Pmat);

```

## D. ROTOR.M

```

function[wn] = rotor(ML,EI,R,n,e,w0,theta,modes,bc);
% rotor(ML,EI,R,n,e,w0,theta,modes,bc)
%
% This function determines the natural frequencies of a rotor of radius 'R' comprised of 'n' elements with
% an offset 'e', mass/length 'ML' and stiffness coefficient 'EI' rotating up to speed 'omegamx' and angle of
% pitch 'theta'. The function will determine the natural frequencies up to the number of 'modes' requested.
% 'bc' sets the boundary conditions, for bc=0 pinned end is assumed, for bc=1 cantilever is assumed 'w0' is
% the rotor normal operating speed.
%
% Dan Hiatt, November 1994

if modes>n
    error('too many modes requested for the number of elements given')
    return
end
omegamx=1.5*w0;
l=(R-e)/n;% element length
% Global matrices initialized
Mf=zeros(2*(n+1),2*(n+1));
M=zeros(2*n,2*n);
Kef=zeros(2*(n+1),2*(n+1));
Ke=zeros(2*n,2*n);
Kcf=zeros(2*(n+1),2*(n+1));
Kc=zeros(2*n,2*n);
Kt=zeros(2*n,2*n);
theta=theta*pi/180;

% fill in the elemental mass and stiffness matrices
m=[156 22 54 -13;22 4 13 -3;54 13 156 -22;-13 -3 -22 4];

```

```

ke=[12 6 -12 6;6 4 -6 2;-12 -6 12 -6;6 2 -6 4];
j=1;

% assemble global mass and elastic stiffness matrices
for i=1:n
    Mf(j:j+3,j:j+3)=Mf(j:j+3,j:j+3)+m;
    Kef(j:j+3,j:j+3)=Kef(j:j+3,j:j+3)+ke;
    if j==2 & bc==0%
        Kef(2,2)=1^3/EI;
    end
    j=2*i+1;
end
if bc==0 % apply boundary conditions to global matrices
    M=Mf(2:2*(n+1),2:2*(n+1))*ML*l/420;
    Ke=Kef(2:2*(n+1),2:2*(n+1))*EI/l^3;
else
    M=Mf(3:2*(n+1),3:2*(n+1))*ML*l/420;
    Ke=Kef(3:2*(n+1),3:2*(n+1))*EI/l^3;
end

% develop the centrifugal stiffness elemental matrix
for k=1:n
    Rprime=e/l;
    d(k)=-(Rprime+(k-1));
    c(k)=Rprime*(n-(k-1))+.5*(n^2-(k-1)^2);
end
j=1;
for i=1:n
    kc(1,1)=6*c(i)/5+3*d(i)/5-6/35;
    kc(3,3)=kc(1,1);
    kc(2,1)=c(i)/10+d(i)/10-1/28;
    kc(1,2)=kc(2,1);
    kc(3,1)=-6*c(i)/5-3*d(i)/5+6/35;
    kc(1,3)=kc(3,1);
    kc(4,1)=c(i)/10+1/70;
    kc(1,4)=kc(4,1);
    kc(2,3)=-kc(2,1);
    kc(3,2)=kc(2,3);
    kc(4,3)=-kc(4,1);
    kc(3,4)=kc(4,3);
    kc(4,2)=-c(i)/30-d(i)/60+1/140;
    kc(2,4)=kc(4,2);
    kc(2,2)=2*c(i)/15+d(i)/30-1/105;
    kc(4,4)=2*c(i)/15+d(i)/10-3/70;
    Kcf(j:j+3,j:j+3)=Kcf(j:j+3,j:j+3)+kc;
    j=2*i+1;
end
for f=1:omegamx+1
    index(f)=f-1;
    if bc==0
        Kc=Kcf(2:2*(n+1),2:2*(n+1))*ML*l*(f-1)^2;
    end
end

```



```

else
    Kc=Kcf(3:2*(n+1),3:2*(n+1))*ML*I*(f-1)^2;
end
% assemble the total stiffness matrix and form the dynamical matrix 'D'
Kt=Ke+Kc;
D=inv(Kt)*M;

% assume a mode shape and use power method to resolve natural frequency
for p=1:modes
    u=ones(max(size(Kc)),1);
    uprev=u;
    for k=1:50
        lambu=D*u;
        u=lambu/max(lambu);
        if u==uprev
            break
        else
            uprev=u;
        end
    end
    wn(p,f)=(inv(max(lambu))-(sin(theta))^2*(f-1)^2)^.5;
    z=max(lambu);
% normalize the mode shape wrt the mass and deflate the 'D' matrix
    c=u'*M*u;
    c=inv(sqrt(c));
    unorm=c*u;
    D=D-z*unorm*unorm'*M;
end
% keep track of multiples of rotor speed
for p=1:modes+1
    np(p,f)=p*(f-1);
end
end
wn/w0;
for p=1:modes
    hold on
    plot(index/w0,wn(p,1:f)/w0);
end
for p=1:modes+1
    plot(index/w0,np(p,1:f)/w0,'r. ');
end
xlabel('Rotational speed ratio omega/omega_0');
ylabel('Natural frequency ratio omega_n/omega_0');
%grid on
%mode(1:n,1)=u(1:2:2*n)/u(2*n-1)

```

## E. RTRTOT.M

```
function[vv] = rtrtot(ML,El,R,n,e,omega,theta,modes,bc,Fp);
```

```

% rtrtot(ML,EI,R,n,e,omega,theta,modes,bc,Fp)
%
% This function determines the displacement Wx, slope, moment and shear of a rotor of radius 'R'
% comprised of 'n' elements with an offset 'e', mass/length 'ML' and stiffness coefficient 'EI' rotating at speed
% 'omegamx' and angle of pitch 'theta'. The function will determine the distributions up to the number of
% 'modes' requested (# modes <= # elements). bc=0 for pinned end and bc=1 for cantilever. Fp is the forcing
% frequency
% Dan Hiatt, November 1994

if modes>n
    error('too many modes requested for the number of elements given')
    return
end
l=(R-e)/n;
Mf=zeros(2*(n+1),2*(n+1));
M=zeros(2*n,2*n);
Kef=zeros(2*(n+1),2*(n+1));
Ke=zeros(2*n,2*n);
Kcf=zeros(2*(n+1),2*(n+1));
Kc=zeros(2*n,2*n);
Kt=zeros(2*n,2*n);
theta=theta*pi/180;
phi=zeros(2*n,modes);
row=.0023769;
chord=1.5;
cla=2*pi; % estimate for 0012 airfoil change later to match
Cf=zeros(2*(n+1),2*(n+1));
C=zeros(2*n,2*n);
Pf=zeros(2*(n+1),1);
P=zeros(2*n,1);

% fill in the mass and stiffness matrices
m=[156 22 54 -13;22 4 13 -3;54 13 156 -22;-13 -3 -22 4];
ke=[12 6 -12 6;6 4 -6 2;-12 -6 12 -6;6 2 -6 4];
k=1;
for i=1:n
    Mf(k:k+3,k:k+3)=Mf(k:k+3,k:k+3)+m;
    Kef(k:k+3,k:k+3)=Kef(k:k+3,k:k+3)+ke;
    if k==2 & bc==0%
        Kef(2,2)=l^3/EI;
    end
    k=2*i+1;
end
if bc==0
    M=Mf(2:2*(n+1),2:2*(n+1))*ML*l/420;
    Ke=Kef(2:2*(n+1),2:2*(n+1))*EI/l^3;
else
    M=Mf(3:2*(n+1),3:2*(n+1))*ML*l/420;
    Ke=Kef(3:2*(n+1),3:2*(n+1))*EI/l^3;
end
end

```

```

% develop the centrifugal stiffness elemental matrix
for k=1:n
    Rprime=e/l;
    d(k)=-(Rprime+(k-1));
    c(k)=Rprime*(n-(k-1))+.5*(n^2-(k-1)^2);
end
k=1;
for i=1:n
    kc(1,1)=6*c(i)/5+3*d(i)/5-6/35;
    kc(3,3)=kc(1,1);
    kc(2,1)=c(i)/10+d(i)/10-1/28;
    kc(1,2)=kc(2,1);
    kc(3,1)=-6*c(i)/5-3*d(i)/5+6/35;
    kc(1,3)=kc(3,1);
    kc(4,1)=c(i)/10+1/70;
    kc(1,4)=kc(4,1);
    kc(2,3)=-kc(2,1);
    kc(3,2)=kc(2,3);
    kc(4,3)=-kc(4,1);
    kc(3,4)=kc(4,3);
    kc(4,2)=-c(i)/30-d(i)/60+1/140;
    kc(2,4)=kc(4,2);
    kc(2,2)=2*c(i)/15+d(i)/30-1/105;
    kc(4,4)=2*c(i)/15+d(i)/10-3/70;
    Kcf(k:k+3,k:k+3)=Kcf(k:k+3,k:k+3)+kc;
    k=2*i+1;
end
f=omega;
if bc==0
    Kc=Kcf(2:2*(n+1),2:2*(n+1))*ML*1*(f)^2;
else
    Kc=Kcf(3:2*(n+1),3:2*(n+1))*ML*1*(f)^2;
end
index=max(size(Kc));

% assemble the total stiffness matrix and form the dynamical matrix 'D'
Kt=Ke+Kc;
D=inv(Kt)*M;

% assume a mode shape and use power method to resolve natural frequency
for p=1:modes
    u=ones(index,1);
    uprev=u;
    for k=1:50
        lambu=D*u;
        u=lambu/max(lambu);
        if u==uprev
            break
        else
            uprev=u;
        end
    end
end

```

```

end
wn(p)=(inv(max(lambu))-(sin(theta))^2*(f)^2)^.5;
z=max(lambu);
% normalize the mode shape wrt the mass and deflate the 'D' matrix
c=u'*M*u;
c=inv(sqrt(c));
unorm=c*u;
D=D-z*unorm*unorm'*M;
phi(1:index,p)=unorm(1:index);
end
wn
phi;

```

```

% develop the aerodynamic damping matrix

```

```

damp=row*cla*f*chord;
k=1;
for i=1:n
    c(1,1)=damp*(3*l^2/70+13*l*(i-1)/70);
    c(2,1)=damp*(11*l^2*(i-1)/420+l^3/120);
    c(3,1)=damp*(9*l*(i-1)/140+9*l^2/280);
    c(4,1)=damp*(-13*l^2*(i-1)/840-l^3/140);
    c(2,2)=damp*(l^3*(i-1)/210+l^4/560);
    c(3,2)=damp*(l^3/120+13*l^2*(i-1)/840);
    c(4,2)=damp*(-l^3*(i-1)/280-l^4/560);
    c(3,3)=damp*(3*l*(i-1)/5+3*l^2/10);
    c(4,3)=damp*(-l^2*(i-1)/20);
    c(4,4)=damp*(l^3*(i-1)/210+l^4/336);
    c(1,2)=c(2,1);
    c(1,3)=c(3,1);
    c(1,4)=c(4,1);
    c(2,3)=c(3,2);
    c(2,4)=c(4,2);
    c(3,4)=c(4,3);
    Cf(k:k+3,k:k+3)=Cf(k:k+3,k:k+3)+c;
    k=2*i+1;
end
if bc==0
    C=Cf(2:2*(n+1),2:2*(n+1));
else
    C=Cf(3:2*(n+1),3:2*(n+1));
end

```

```

% develop the forcing matrix

```

```

k=1;
for i=1:n
    p(1,1)=l/2;
    p(2,1)=l^2/12;
    p(3,1)=l/2;
    p(4,1)=-l^2/12;
    Pf(k:k+3,1)=Pf(k:k+3,1)+p*40; % 40 lb/ft load distribution
    k=2*i+1;
end

```

```

end
q=zeros(1:index,1);
if bc==0
    P=Pf(2:2*(n+1));
else
    P=Pf(3:2*(n+1));
end

% solve the non-homogeneous equations of motion in modal coordinates
for i=1:modes
    rl=(wn(i)^2-Fp^2);
    Cbar=phi(1:index,i)*C*phi(1:index,i);
    im=j*f*Cbar;
    P0=phi(1:index,i)*P;
    z=inv(rl+im);
    q0=z*P0;

% convert back to real coordinates
    q=q+phi(1:index,i)*q0;
end
q=real(q);
if bc==0
    qf(2:index+1)=q;
else
    qf(3:index+2)=q;
end
qf(2:2:2*n+2)=qf(2:2:2*n+2)/l;
q=qf;
k=1;
r=1;
Wx(1)=0;
% apply interpolating polynomials to displacement, slope, moment and shear
for i=1:n
    c1=(6/l^3)*(2*q(k)+l*q(k+1)-2*q(k+2)+l*q(k+3));
    c2=(2/l^2)*(-3*q(k)-2*l*q(k+1)+3*q(k+2)-l*q(k+3));
    c3=q(k+1);
    if i==n
        for x=0:l/10:l
            w(1)=1-3*(x/l)^2+2*(x/l)^3;
            w(2)=(x-2*l*(x/l)^2+l*(x/l)^3);
            w(3)=3*(x/l)^2-2*(x/l)^3;
            w(4)=(l*(x/l)^3-l*(x/l)^2);
            Wx(r)=w*qf(k:k+3)';
            dWx(r)=(.5*c1*x^2+c2*x+c3);
            d2Wx(r)=EI*(c1*x+c2);
            d3Wx(r)=EI*(c1);
            T(r)=(ML*f^2*R^2-ML*f^2*((i-1)*l+e+x)^2)/2;
            index(r)=(e+(i-1)*l+x)/R;
            r=r+1;
        end
    end
else

```

```

for x=0:l/10:9*l/10
    w(1)=1-3*(x/l)^2+2*(x/l)^3;
    w(2)=(x-2*l*(x/l)^2+l*(x/l)^3);
    w(3)=3*(x/l)^2-2*(x/l)^3;
    w(4)=(l*(x/l)^3-l*(x/l)^2);
    Wx(r)=w*ql(k:k+3)';
    dWx(r)=(.5*c1*x^2+c2*x+c3);
    d2Wx(r)=EI*(c1*x+c2);
    d3Wx(r)=EI*(c1);
    T(r)=(ML*f^2*R^2-ML*f^2*((i-1)*l+e+x)^2)/2;
    index(r)=(e+(i-1)*l+x)/R;
    r=r+1;
end
end
k=k+2;
end
for i=1:max(size(T))
    shear(i)=-(d3Wx(i)-T(i)*dWx(i));
end
subplot(2,2,4)
plot(index,Wx*12,'g')
axis([0 1 min(Wx)*12 max(Wx)*12])
%title('DISPLACEMENT vs r/R')
xlabel('r/R')
ylabel('Displacement, in')
hold on, grid on
subplot(2,2,3)
plot(index,dWx*57.3)
axis([0 1 min(dWx)*57.3 max(dWx)*57.3])
%title('SLOPE vs r/R')
xlabel('r/R')
ylabel('Slope, deg')
hold on, grid on
subplot(2,2,2)
plot(index,d2Wx,'r')
y1=max(d2Wx)*1.05;
y2=min(d2Wx)*1.05;
axis([0 1.0 y2 y1]);
%title('MOMENT vs r/R')
xlabel('r/R')
ylabel('Moment, ft-lb')
grid on, hold on
subplot(2,2,1)
plot(index,shear,'r')
%title('SHEAR vs r/R')
xlabel('r/R')
ylabel('Shear, lb')
grid on, hold on

```

## F. BLDREV.M

% Myklestad-Thomson based blade dynamic response program which  
% incorporates articulated and cantilever root conditions. Also included  
% are lag damping, coupled flap and lag responses due to twist and coupling  
% due to coriolis force. Not included are torsional responses.

%

% LT DAN HIATT, FEBRUARY 1995

clc

disp(' ')

disp(' ')

disp('

\*\*\* UTILIZING THE MYKLESTAD-THOMSON METHOD TO SOLVE \*\*\*')

disp('

\*\*\* FOR FORCED BLADE DYNAMIC RESPONSE \*\*\*')

%acdat='danh60' %%%%%%%%%% diagnostics

%blddat='danh1' %%%%%%%%%% diagnostics

eval(['load ',acdat ])

eval(['load ',[acdat '\_p']])

eval(['load ',blddat])

% enter lag damping information

disp(' ')

disp(' ')

disp('Enter the value for lag damper damping coefficient (lb-ft/sec) ')

disp('enter zero if no lag damper is installed')

Cthta=input(' >>');

modes=11;

if exist('Ex')==1

    Elx=Ex.\*Ixx;

    Ely=Ey.\*Iyy;

end

thetao=thetao\*pi/180; % collective at .7\*R

delr=(R-e)/nbe;

deltwst=twist/nbe; % change in blade angle due to twist per element

Beta0=thetao+twist/(R-e)\*.3\*R;

rho=.0023778;

betao=4.5/57.3;

for k=nbe:-1:1,

    rn((nbe+1)-k)=e+k\*delr-delr/2;% find radius at midpoint of each element

end

rn=(rn); % element radius in inches

Xout=[rn]./R; % index for plotting results versus r/R

lsn=(rn(3)-rn(4))\*12; % all element lengths are the same (inches)

mn=Wn\*lsn/(32.174); % concentrated mass of an element (slug)

disp(' ')

```

disp('          *** CALCULATING THE STEADY AND TEN HARMONIC ***')
disp('          *** DIFFERENTIAL THRUST ELEMENTS ***')

% reverse columns so radius goes from tip to root
Fn=dT;
Fn=[((Fn(:,1)+Fn(:,2))/2) Fn(:,3:nbe+1));
Fn=fliplr(dT);
Dn=dD;
Dn=[((Dn(:,1)+Dn(:,2))/2) Dn(:,3:nbe+1));
Dn=fliplr(dD);
% calculate harmonics:
% setting up rows of harmonics and columns of blade stations

%%%%%%%%%%%%%% diagnostics section
%clear Fn
%for i=0:35
%for k=1:21
% Fn(i+1,k)=10*(k-1)*sin(3*i*10*pi/180);
% Dn(i+1,k)=10*(k-1)*sin(3*i*10*pi/180);
%end
%end
%Dn=fliplr(Dn);
%Dn=[((Dn(:,1)+Dn(:,2))/2) Dn(:,3:nbe+1));
%Fn=fliplr(Fn);
%Fn=[((Fn(:,1)+Fn(:,2))/2) Fn(:,3:nbe+1));
%%%%%%%%%%%%%%

% calculate steady state term for lift and drag
for j=1:nbe
    dFo(j)=0;
    dDo(j)=0;
    for k=1:naz
        dFo(j)=dFo(j)+Fn(k,j);
        dDo(j)=dDo(j)+Dn(k,j);
    end
end
dFo=dFo./naz;
dDo=dDo./naz;

%Calculate dFn and dDn (sin term) harmonic matrices
for i=1:10,
    for j=1:nbe,
        dFn(i,j)=0;
        dDn(i,j)=0;
        for k=1:naz,
            Fcum=Fn(k,j)*sin(i*psi(k)/57.3);
            dFn(i,j)=dFn(i,j)+Fcum;
            Dcum=Dn(k,j)*sin(i*psi(k)/57.3);
            dDn(i,j)=dDn(i,j)+Dcum;
        end
        dFn(i,j)=dFn(i,j)/(naz/2);
    end
end

```



```

    dDn(i,j)=dDn(i,j)/(naz/2);
end
end

% calculate dfn and ddn (cos term) harmonic matrices
for i=1:10,
    for j=1:nbe,
        dfn(i,j)=0;
        ddn(i,j)=0;
        for k=1:naz,
            fcum=Fn(k,j)*cos(i*psi(k)/57.3);
            dfn(i,j)=dfn(i,j)+fcum;
            dcum=Dn(k,j)*cos(i*psi(k)/57.3);
            ddn(i,j)=ddn(i,j)+dcum;
        end
        dfn(i,j)=dfn(i,j)/(naz/2);
        ddn(i,j)=ddn(i,j)/(naz/2);
    end
end

disp(' ')
disp('*** CALCULATING ELEMENT AERO DAMPING, TENSION & TWIST COUPLING ***')

% Calculating the flatwise aerodynamic damping, Cn, the tension, Tn,
% and the combined collective and twist, Beta, for each element
cblade2=cblade;
Cn=zeros(1,nbe);
Tn=zeros(1,nbe);
Tn(1)=mn(1)*omega^2*rn(1);% calculating tension since it is independent
Beta(1)=Beta0;% collective setting and twist combined value at rotor tip

for k=1:nbe-1,
    Cn(k)=a*cblade2(k+1)*(rn(k)-rn(k+1)).*5*rho*omega*rn(k);
    Tn(k+1)=Tn(k) + mn(k+1)*omega^2*rn(k+1); % element tension vector
    Beta(k+1)=Beta0-k*deltwst; % collective and twist combined at element
end

% this section develops the twist coupling terms
for k=1:nbe
    vn(k)=lsn/Ely(k); % flatwise slope due to moment
    un(k)=.5*lsn^2/Ely(k); % flatwise slope due to load
    gn(k)=lsn^3/(3*Ely(k)); % flatwise deflection due to load
    Vn(k)=lsn/Elx(k); % chordwise slope due to moment
    Un(k)=.5*lsn^2/Elx(k); % chordwise slope due to load
    Gn(k)=lsn^3/(3*Elx(k)); % chordwise deflection due to load
    vzz(k)=Vn(k)*(sin(Beta(k)))^2+vn(k)*(cos(Beta(k)))^2;
    uzz(k)=Un(k)*(sin(Beta(k)))^2+un(k)*(cos(Beta(k)))^2;
    gzz(k)=Gn(k)*(sin(Beta(k)))^2+gn(k)*(cos(Beta(k)))^2;
    Vxx(k)=vn(k)*(sin(Beta(k)))^2+Vn(k)*(cos(Beta(k)))^2;
    Uxx(k)=un(k)*(sin(Beta(k)))^2+Un(k)*(cos(Beta(k)))^2;
    Gxx(k)=gn(k)*(sin(Beta(k)))^2+Gn(k)*(cos(Beta(k)))^2;

```

```

Vxz(k)=(Vn(k)-vn(k))*sin(Beta(k))*cos(Beta(k));
Uxz(k)=(Un(k)-un(k))*sin(Beta(k))*cos(Beta(k));
Gxz(k)=(Gn(k)-gn(k))*sin(Beta(k))*cos(Beta(k));
vzx(k)=Vxz(k); uzx(k)=Uxz(k); gzx(k)=Gxz(k);
end

clear vn un gn Un Vn Gn Beta % clean up workspace

disp(' ')
disp('*** CALCULATING THE XFER MATRICES FOR THE STEADY AND 10 HARMONIC ***')
disp(' *** BLADE RESPONSES ALONG THE LENGTH OF THE BLADE ***')

j=sqrt(-1);

for i=0:10;
    omegae=omega*i; % steady and ten harmonic excitation frequencies

    for kk=0:4 % pass thru the eqns 5 times to resolve the 4 root unknowns
        % thetai, thetaf, Z and X in terms of the tip values and the
        % response terms associated with force inputs
        thtprvf=0;
        thtprve=0;
        Tprv=0;
        Zprv=0;
        Xprv=0;
        Sprvf=0; % boundary condition
        Sprve=0; % boundary condition
        Mprvf=0; % boundary condition
        Mprve=0; % boundary condition

        if kk==1
            thtprvf=1; % solving for theta flatwise coeff
        elseif kk==2
            Zprv=1; % solving for flatwise deflection coeff
        elseif kk==3
            thtprve=1; % solving for edgewise theta coeff
        elseif kk==4
            Xprv=1; % solving for edgewise deflection coeff
        end % (endif kk)

        for k=1:nbe,
            thtfl(k)=thtprvf*(1+Tprv*uzz(k))+Tprv*thtprve*uzx(k)-Mprvf*vzz(k)...
                -Mprve*vzx(k)-Sprvf*uzz(k)-Sprve*uzx(k);
            thtel(k)=thtprve*(1+Tprv*Uxx(k))+Tprv*thtprvf*Uxz(k)-Mprve*Vxx(k)...
                -Mprvf*Vxz(k)-Sprve*Uxx(k)-Sprvf*Uxz(k);
            Zl(k)=Zprv-thtfl(k)*lsn+Tprv*(thtprvf*gzx(k)+thtprve*gzx(k))...
                -Mprvf*uzz(k)-Mprve*uzx(k)-Sprvf*gzx(k)-Sprve*gzx(k);
            Xl(k)=Xprv-thtel(k)*lsn+Tprv*(thtprve*Gxx(k)+thtprvf*Gxz(k))...
                -Mprve*Uxx(k)-Mprvf*Uxz(k)-Sprve*Gxx(k)-Sprvf*Gxz(k);
            Mfl(k)=Mprvf+Sprvf*lsn-Tprv*(Zprv-Zl(k));
        end
    end
end

```

```

Me1(k)=Mprve+Sprve*lsn-Tprv*(Xprv-X1(k));

% this section determines which forces to apply for each harmonic
if i==0 & kk==0
    alphaf(i+1,k)=dFo(k);
    alphae(i+1,k)=dDo(k);
elseif i~=0 & kk==0
    alphaf(i+1,k)=dFn(i,k)+j*dfn(i,k);
    alphae(i+1,k)=dDn(i,k)+j*ddn(i,k);
end % (endif)

% this section adds the determined forces to the shear component
if kk==0
    Sf1(k)=Sprvf+(mn(k)*omegae^2-j*omegae*Cn(k))*Z1(k)/12+alphaf(i+1,k);
    Se1(k)=Sprve+mn(k)*(omegae^2+omegae^2)*X1(k)/12+alphae(i+1,k)...
        +2*j*omegae*omegae*mn(k)*Z1(k)*betao;
else
    Sf1(k)=Sprvf+(mn(k)*omegae^2-j*omegae*Cn(k))*Z1(k)/12;
    Se1(k)=Sprve+mn(k)*(omegae^2+omegae^2)*X1(k)/12+2*j*omegae*omegae*mn(k)...
        *Z1(k)*betao;
end % endif

% this section renames the current values for looping
thtprvf=thtfl(k);
thtprve=thtel(k);
Tprv=Tn(k);
Zprv=Z1(k);
Xprv=X1(k);
Sprvf=Sf1(k);
Sprve=Se1(k);
Mprvf=Mf1(k);
Mprve=Me1(k);

end % (end for k=1:nbe)

if bc==1 % assemble the soln matrix for articulated blades using root b.c.
    P(1:4,kk+1)=[Mf1(nbe);Z1(nbe);Me1(nbe)-j*Cthta*omegae;X1(nbe)];
else % assemble the soln matrix for cantilever blades using root b.c.
    P(1:4,kk+1)=[thtfl(nbe);Z1(nbe);thtel(nbe);X1(nbe)];
end

end % (end for kk=0:4)

% this section solves the eqn Ax+b=0 by setting x=inv(A)*(-b)
alphart=-P(:,1); % this is the b term or force values
P=P(:,2:5); % this is the A term or response values
tipbc=inv(P)*alphart;

% this section re-initializes the tip conditions with the known values
% then passes thru the eqns determining blade response at each station
thtprvf=tipbc(1);

```

```

thtprve=tipbc(3);
Tprv=0;
Zprv=tipbc(2);
Xprv=tipbc(4);
Sprvf=0; % boundary condition
Sprve=0; % boundary condition
Mprvf=0; % boundary condition
Mprve=0; % boundary condition
for k=1:nbe,
    thtf(i+1,k)=thtprvf*(1+Tprv*uzz(k))+Tprv*thtprve*uzz(k)-Mprvf*vzz(k)...
        -Mprve*vzx(k)-Sprvf*uzz(k)-Sprve*uzz(k);
    thte(i+1,k)=thtprve*(1+Tprv*Uxx(k))+Tprv*thtprvf*Uxz(k)-Mprve*Vxx(k)...
        -Mprvf*Vxz(k)-Sprve*Uxx(k)-Sprvf*Uxz(k);
    Z(i+1,k)=Zprv-thtf(i+1,k)*lsn+Tprv*(thtprvf*gzz(k)+thtprve*gzx(k))...
        -Mprvf*uzz(k)-Mprve*uzz(k)-Sprvf*gzz(k)-Sprve*gzx(k);
    X(i+1,k)=Xprv-thte(i+1,k)*lsn+Tprv*(thtprve*Gxx(k)+thtprvf*Gxz(k))...
        -Mprve*Uxx(k)-Mprvf*Uxz(k)-Sprve*Gxx(k)-Sprvf*Gxz(k);
    Mf(i+1,k)=Mprvf+Sprvf*lsn-Tprv*(Zprv-Z(i+1,k));
    Me(i+1,k)=Mprve+Sprve*lsn-Tprv*(Xprv-X(i+1,k));
    Sf(i+1,k)=Sprvf+(mn(k)*omegae^2-j*omegae*Cn(k))*Z(i+1,k)/12+alphaf(i+1,k);
    Se(i+1,k)=Sprve+mn(k)*(omegae^2+omega^2)*X(i+1,k)/12+alphae(i+1,k)...
        +2*j*omegae*omegae*mn(k)*Z(i+1,k)*betao;
    thtprvf=thtf(i+1,k);
    thtprve=thte(i+1,k);
    Tprv=Tn(k);
    Zprv=Z(i+1,k);
    Xprv=X(i+1,k);
    Sprvf=Sf(i+1,k);
    Sprve=Se(i+1,k);
    Mprvf=Mf(i+1,k);
    Mprve=Me(i+1,k);
end

% set the required tip conditions for shear and moment
Sf(i+1,1)=0;
Mf(i+1,1)=0;

end % end (for i=0:10)

% call the output program with the required variables
output

```

## G. OUTPUT.M

```

%function[vv]=output(Xout,Z,X,thtf,thte,Mf,Me,Sf,Se,naz,psi,Ely,Elx,Wn,modes,rn)

%output(Xout,Z,X,thtf,thte,Mf,Me,Sf,Se,naz,psi,Ely,Elx,Wn,modes,rn)
%
% Function called to view outputs from BLDREV.M
%

```

% by Lt Juan D. Cuesta, REVISED by LT DAN HIATT  
% September 1994, REVISED FEBRUARY 1995

```
view=1;
while view > 0,
disp(' ')
disp('      *** BLADE DYNAMICS OUTPUT MENU ***')
disp(' ')
disp('      CHOOSE WHICH OUTPUT OPTION YOU WOULD LIKE')
disp(' ')
disp(' 1. View the steady and first ten harmonic responses')
disp(' ')
disp(' 2. View a mesh plot of the Shear, Moment, Slope and Deflection')
disp('      at all azimuth positions')
disp(' ')
disp(' 3. View the Shear, Moment, Slope and Shear at a specific')
disp('      azimuth position')
disp(' ')
disp(' 4. View the total response at a given radius r')
disp(' ')
disp(' 5. View the stiffness (EI) and weight distribution')
disp(' ')
disp(' 0. Exit')
disp(' ')
disp(' *** FOR A PRINTOUT CHOOSE THE "File" OPTION IN THE DESIRED GRAPH WINDOW ***')
view=input('    >> ');
if view==0
    return
end
disp(' ')
disp(' 1. View the flatwise response')
disp(' ')
disp(' 2. View the edgewise response')
disp(' ')
subview=input('    >> ');
if subview==1
    var1=Sf;
    var2=Mf;
    var3=thtf;
    var4=Z;
    var5=Ely;
elseif subview==2
    var1=Se;
    var2=Me;
    var3=thte;
    var4=X;
    var5=Elx;
end
```

% viewing the steady and the first ten harmonic responses

```

if view==1,
disp(' ')
for i=3:5%0:modes-1,
figure(i+1)
subplot(2,2,1)
plot(Xout,real(var1(i+1,:))+imag(var1(i+1,:)),'r');grid
%title(sprintf('SHEAR ,%3.0f HARMONIC RESPONSE',i))
xlabel('r/R')
ylabel('SHEAR (LBS)')

subplot(2,2,2)
plot(Xout,real(var2(i+1,:))+imag(var2(i+1,:)),'b');grid
xlabel('r/R')
ylabel('MOMENTS (IN-LBS)')
%title(sprintf('.....MOMENT, %3.0f HARMONIC RESPONSE',i))

subplot(2,2,3)
plot(Xout,(real(var3(i+1,:)))*180/pi+(imag(var3(i+1,:)))*180/pi,'b');grid
%title(sprintf('SLOPE,%3.0f HARMONIC RESPONSE',i))
xlabel('r/R')
ylabel('SLOPE (DEG)')

subplot(2,2,4)
plot(Xout,real(var4(i+1,:))+imag(var4(i+1,:)),'b');grid
%title(sprintf('DEFLECTION,%3.0f HARMONIC RESPONSE',i))
xlabel('r/R')
ylabel('DISPLACEMENT (IN)')
end
end

% viewing the mesh plot for the total response
if view==2,
figure(1)
YoutS=zeros(1,20);
YoutM=zeros(1,20);
YoutTh=zeros(1,20);
YoutY=zeros(1,20);
for j=1:length(psi),
Yout3=real([var1(1,:);var2(1,:);var3(1,:);var4(1,:)]); %steady state
for i=1:10, %harmonics
Yout2=[var1(i+1,:);var2(i+1,:);var3(i+1,:);var4(i+1,:)];
Ynew=real(Yout2).*sin(i*psi(j)/57.3)+imag(Yout2).*cos(i*psi(j)/57.3);
Yout3=Yout3+Ynew;
end
YoutS(j,:)=Yout3(1,:);
YoutM(j,:)=Yout3(2,:);
YoutTh(j,:)=Yout3(3,:)*180/pi;
YoutY(j,:)=Yout3(4,:);
end

%subplot(2,2,1)

```

```

mesh(Xout,psi,YoutS)
%title('TOTAL RESPONSE SHEAR PLOT')
xlabel('r/R')
ylabel('azimuth, deg')
zlabel('SHEAR (LBS)')

figure(2)
%subplot(2,2,2)
mesh(Xout,psi,YoutM)
%title('TOTAL RESPONSE MOMENT PLOT')
xlabel('r/R')
ylabel('azimuth, deg')
zlabel('MOMENTS (IN-LBS)')

figure(3)
%subplot(2,2,3)
mesh(Xout,psi,YoutTh)
%title('TOTAL RESPONSE SLOPE PLOT')
xlabel('r/R')
ylabel('azimuth, deg')
zlabel('SLOPE (DEG)')

figure(4)
%subplot(2,2,4)
mesh(Xout,psi,YoutY)
%title('TOTAL RESPONSE DEFLECTION PLOT')
xlabel('r/R')
ylabel('azimuth, deg')
zlabel('DISPLACEMENT (IN)')
end

% viewing the total response at a specific azimuth
if view==3
    clc
    flag=1;
    pic=0;
    while flag > 0,
        pic=pic+1;
        disp(' ')
        az=input('Enter the azimuth angle at which you wish to see the total response (deg): ');
        Yout3=real([var1(1,:);var2(1,:);var3(1,:);var4(1,:)]); %steady state
        for i=1:modes-1, %harmonics
            Yout2=[var1(i+1,:);var2(i+1,:);var3(i+1,:);var4(i+1,:)];
            Ynew=real(Yout2).*sin(i*az/57.3)+imag(Yout2).*cos(i*az/57.3);
            Yout3=Yout3+Ynew;
        end
        figure(pic)
        subplot(2,2,1)
        plot(Xout,Yout3(1,:),'b-');grid
        % title(sprintf('TOTAL RESPONSE SHEAR AT%3.0f DEG',az))
        xlabel('r/R')

```

```

ylabel('SHEAR (LBS)')

subplot(2,2,2)
plot(Xout,Yout3(2,:),'b-');grid
% title(sprintf('TOTAL RESPONSE MOMENT AT%3.0f DEG',az))
xlabel('r/R')
ylabel('MOMENTS (IN-LBS)')

subplot(2,2,3)
plot(Xout,Yout3(3,:)*180/pi,'b-');grid
% title(sprintf('TOTAL RESPONSE SLOPE AT%3.0f DEG',az))
xlabel('r/R')
ylabel('SLOPE (DEG)')

subplot(2,2,4)
plot(Xout,Yout3(4,:),'b-');grid
% title(sprintf('TOTAL RESPONSE DEFLECTION AT%3.0f DEG',az))
xlabel('r/R')
ylabel('DISPLACEMENT (IN)')

disp(' ')
disp('Do you want to see another azimuth angle?')
flag=input(' 0) No    1) Yes >>');
end
end

% viewing the total response at any r
if view==4,
    clc
    disp('      Enter the radius desired in % of R')
    disp('example--0.75--corresponds to the blade station closest to 75% of blade length')
    disp(' ')
    rad=input('    >>');
    sta=round((1-rad)*max(size(Xout)));

figure(1)
YoutS=0;
YoutM=0;
YoutTh=0;
YoutY=0;
for j=1:length(psi),
    Yout3=real([var1(1,sta);var2(1,sta);var3(1,sta);var4(1,sta)]); %steady state
    for i=1:10, %harmonics
        Yout2=[var1(i+1,sta);var2(i+1,sta);var3(i+1,sta);var4(i+1,sta)];
        Ynew=real(Yout2).*sin(i*psi(j)/57.3)+imag(Yout2).*cos(i*psi(j)/57.3);
        Yout3=Yout3+Ynew;
    end
    YoutS(j,1)=Yout3(1,1);
    YoutM(j,1)=Yout3(2,1);
    YoutTh(j,1)=Yout3(3,1);
    YoutY(j,1)=Yout3(4,1);

```



```

end

subplot(2,2,1)
plot(psi, YoutS), grid on
%title(sprintf('TOTAL RESPONSE SHEAR PLOT at %3.2f Of R', rad))
xlabel('azimuth (deg)')
ylabel('SHEAR (LB)')

subplot(2,2,2)
plot(psi, YoutM), grid on
%title(sprintf('TOTAL RESPONSE MOMENT PLOT at %3.2f Of R', rad))
xlabel('azimuth (deg)')
ylabel('MOMENTS (IN-LBS)')

subplot(2,2,3)
plot(psi, YoutTh*180/pi), grid on
%title(sprintf('TOTAL RESPONSE SLOPE PLOT at %3.2f Of R', rad))
xlabel('azimuth (deg)')
ylabel('SLOPE (DEG)')

subplot(2,2,4)
plot(psi, YoutY), grid on
%title(sprintf('TOTAL RESPONSE DEFLECTION PLOT at %3.2f Of R', rad))
xlabel('azimuth (deg)')
ylabel('DISPLACEMENT (IN)')

end

if view==5
    clc
    if subview==1
        axis=('FLATWISE');
    elseif subview==2
        axis=('EDGEWISE');
    end
    subplot(2,1,1)
    plot(rn*12, var5./1e6); grid
    title([axis, ' BLADE ELASTIC STIFFNESS AND WEIGHT DISTRIBUTIONS'])
    xlabel('BLADE STATION, IN')
    ylabel('STIFFNESS, EI, LB*IN^2x10^6')

    subplot(2,1,2)
    plot(rn*12, Wn); grid
    % title(['WEIGHT DISTRIBUTION FOR BLADE',])
    xlabel('BLADE STATION, IN')
    ylabel('WEIGHT, LB/IN')
end
end

```

## H. ADVFLOQ.M

% Developed by: DAN HIATT 28APR95

%

% This routine develops the Floquet state transition matrix for the single rotor blade in rotating coordinates. The references for its development include Biggers, Hohenemser and Hammond. The input variables are the lock number (ga), the tip loss factor (B), the flapping natural frequency ratio (nu) and the advance ratio (mu). The output matrix (alpha) contains the blade frequency and damping information. First take the eigenvalues of the alpha matrix then the natural log of these values. The resulting numbers are then divided by the period as follows:

% alpha = 0.1334 0.0013

% 0.0382 0.0677

% eig(alpha) = 0.1342

% 0.0669

% log(ans)/2/pi = -0.3196

% -0.4304

% This routine is designed to iterate through on advance ratio and return values for damping and frequency ratios. A word of caution: the frequency ratios will be associated with an integer multiple of 1/2 per rev thus the actual (absolute)value could be as listed, 1-the (absolute)value listed or 1+the (abs)value listed .

clear

nu=1.1;

ga=5;

B=1;

del3=0; % pitch-flap coupling angle

h=2\*pi/360;

Kp=tan(del3);%\*pi/180); % pitch-flap coupling feedback

Kr=0; % flap rate feedback gain

alpha=zeros(2);

ind=1;

% loop for several values of mu and the given blade conditions

for indx=1:5:101

mu=(indx)/50;

index(ind)=mu;

% one pass for each initial condition

for j=1:2

if j==1

bt10=1;

bt20=0;

else

bt10=0;

bt20=1;

end

% pass through the rotor azimuth

for psi=0:h:2\*pi-h

k1=-h\*bt20;

[Mbdot,Mb,Mthta]=coeff(psi,mu,B);

l1=-h\*ga\*((Mbdot+Kr\*Mthta)\*bt20-(Mb+Kp\*Mthta+nu^2/ga)\*bt10);

```

bt2=bt20+l1/2;
bt1=bt10+k1/2;
k2=-h*(bt2);
psi=psi+h/2;
[Mbdot,Mb,Mthta]=coeff(psi,mu,B);
l2=-h*ga*((Mbdot+Kr*Mthta)*bt2-(Mb+Kp*Mthta+nu^2/ga)*bt1);

bt2=bt20+l2/2;
bt1=bt10+k2/2;
k3=-h*(bt2);
[Mbdot,Mb,Mthta]=coeff(psi,mu,B);
l3=-h*ga*((Mbdot+Kr*Mthta)*bt2-(Mb+Kp*Mthta+nu^2/ga)*bt1);

bt2=bt20+l3;
bt1=bt10+k3;
k4=-h*(bt2);
psi=psi+h/2;
[Mbdot,Mb,Mthta]=coeff(psi,mu,B);
l4=-h*ga*((Mbdot+Kr*Mthta)*bt2-(Mb+Kp*Mthta+nu^2/ga)*bt1);

% perform the integration step forward
bt1=bt10+(k1+2*k2+2*k3+k4)/6;
bt2=bt20+(l1+2*l2+2*l3+l4)/6;

bt10=bt1;
bt20=bt2;
end
% fill in the columns of the floquet transition matrix
alpha(1,j)=bt1;
alpha(2,j)=bt2;
end
indx

% determine the eigenvalues of the transition matrix and the
% damping and frequency
eigval=eig(alpha);
logval=log(eigval)/2/pi;
logmat(ind,1)=logval(1);
logmat(ind,2)=logval(2);
damp(ind,1)=real(logval(1));
damp(ind,2)=real(logval(2));
freq(ind,1)=imag(logval(1));
freq(ind,2)=imag(logval(2));
ind=ind+1;
end
subplot(2,1,1); plot(index,damp); grid on
title('FLOQUET ANALYSIS - ROTOR FLAPPING STABILITY')
xlabel('Advance ratio')
ylabel('Modal damping')
subplot(2,1,2); plot(index,1+abs(freq)); grid on

```

```

xlabel('Advance ratio')
ylabel('Modal frequency (cycles/sec)')

return

subplot(2,1,1);hold on; plot(index,damp);plot(index,damp1,'-.');plot(index,damp2,'--'); grid on
title('FLOQUET ANALYSIS - ROTOR FLAPPING STABILITY')
xlabel('Advance ratio')
ylabel('Modal damping');hold off
subplot(2,1,2);hold on; plot(index,1+abs(freq));plot(index,1+abs(freq1),'-.');plot(index,1+abs(freq2),'--');grid on
xlabel('Advance ratio')
ylabel('Modal frequency ratio')

```

## I. COEFF.M

```

function[Mbdot,Mb,Mthta]=coeff(psi,mu,B)

% coeff(psi,mu,B)
%
% This function returns the moment coefficients for the floquet analysis function (FLOQ) based on the
% region of disk.
% Dan Hiatt May 1995

if psi >= 2*pi
    psi = 0;
end

if 0 <= psi & psi < pi
    Mbdot=(B^4/8+B^3*mu*sin(psi)/6);
    Mb=(B^3*mu*cos(psi)/6+B^2*(mu^2)*sin(2*psi)/8);
    Mthta=(B^4/8+B^3*mu*sin(psi)/3+B^2*(mu*sin(psi))^2/4);

elseif B/mu >= 1 & pi <= psi & psi < 2*pi
    Mbdot=(B^4/8+B^3*mu*sin(psi)/6+(mu*sin(psi))^4/12);
    Mb=(mu*cos(psi)*B^3/6+B^2*mu^2*sin(2*psi)/8-mu*cos(psi)*(mu*sin(psi))^3/6);
    Mthta=(B^4/8+B^3*mu*sin(psi)/3+(B^2*mu*sin(psi))^2/4-(mu*sin(psi))^4/12);

elseif B/mu < 1
    if psi >= pi & psi < pi+asin(B/mu)
        Mbdot=(B^4/8+B^3*mu*sin(psi)/6+(mu*sin(psi))^4/12);
        Mb=mu*cos(psi)*(B^3/6+B^2*mu*sin(psi)/4-(mu*sin(psi))^3/6);
        Mthta=(B^4/8+B^3*mu*sin(psi)/3+(B^2*mu*sin(psi))^2/4-(mu*sin(psi))^4/12);
    elseif psi > 2*pi-asin(B/mu) & psi < 2*pi
        Mbdot=(B^4/8+B^3*mu*sin(psi)/6+(mu*sin(psi))^4/12);
        Mb=mu*cos(psi)*(B^3/6+B^2*mu*sin(psi)/4-(mu*sin(psi))^3/6);
        Mthta=(B^4/8+B^3*mu*sin(psi)/3+(B^2*mu*sin(psi))^2/4-(mu*sin(psi))^4/12);
    else
        Mbdot=-(B^4/8+B^3*mu*sin(psi)/6);
    end
end

```

```

Mb=-(B^3*mu*cos(psi)/6+B^2*(mu^2)*sin(2*psi)/8);
Mthta=-(B^4/8+B^3*mu*sin(psi)/3+B^2*(mu*sin(psi))^2/4);
end
end

```

## J. CCGRES.M

% developed by DAN HIATT MAY 2 1995

%

% This routine returns a plot of the eigenvalues related to the modal damping and frequency of a 4 bladed rotor in rotating coordinates. The routine assumes an isotropic hub and can accept either isotropic or 1 lag damper out conditions. Motion of the rotor hub is allowed and the equations of motion have been reduced to the constant coefficient case. This routine serves a useful role in determining the minimum lag damping required to ensure stability.

```

N=4; % # blades
e=1; % lag hinge offset (ft)
Sb=65; % blade mass moment (slug-ft)
Ib=800; % blade mass moment of inertia (slug-ft^2)
mb=6.5; % blade mass (slug)
k=0; % lag spring (ft-lb/rad)
c=[3000 3000 3000 0] % lag damper (ft-lb-sec/rad);
mh=550; % hub mass (slug)
kh=85000; % hub spring (lb/ft)
ch=3500; % hub damper (lb-sec/ft)

```

% calculate various constants

```

nusq=e*Sb/Ib;
omsq=k/Ib;
eta=c./Ib;
nuhsq=Sb/(mh+N*mb);
omhsq=kh/(mh+N*mb);
etah=ch/(mh+N*mb);
rpmmx=400;

```

% loop for rpm

```

indx=1;
for i=0:2:450
    rpm(indx)=i/rpmmx;
    hub(indx)=sqrt(nusq);
    zero(indx)=0;
    omega=i*2*pi/60;
    clh=2*nuhsq*omega;
    cl=2*nusq*omega/e;
    k1h=nuhsq*omega^2;
    k1=nusq*(omega^2)/e;
    const=-omsq-nusq*omega^2;
    cnsth=omega^2-omhsq;
    cnst2=nuhsq*omega^2;
    A2=[ 1      0      0      0      0      nusq/e;

```

```

0      1      0      0      -nusq/e 0;
0      0      1      0      0      -nusq/e;
0      0      0      1      nusq/e 0;
0      -nuhsq 0      nuhsq 1      0;
nuhsq 0      -nuhsq 0      0      1];
%
A1=[ -eta(1) 0      0      0      -c1      0;
0      -eta(2) 0      0      0      -c1;
0      0      -eta(3) 0      c1      0;
0      0      0      -eta(4) 0      c1;
c1h      0      -c1h      0      -etah      2*omega;
0      c1h      0      -c1h      -2*omega -etah];
%
% for 1 damper inop calculation, replace one (-eta) value on the
% A1 diagonal with 0.
%
A0=[ const 0      0      0      0      k1;
0      const 0      0      -k1      0;
0      0      const 0      0      -k1;
0      0      0      const k1      0;
0      -cnst2 0      cnst2 cnsth omega*etah;
cnst2 0      -cnst2 0      -omega*etah cnsth];
%
TL=inv(A2)*A1;
TR=inv(A2)*A0;
A(1:6,1:6)=TL;
A(1:6,7:12)=TR;
A(7:12,1:6)=eye(6);
eigenval(:,indx)=eig(A);
indx=indx+1;
end
subplot(2,1,1); plot(rpm,real(eigenval(1:2:12,:)),'k. ');hold on; plot(rpm,zero,'r');
title('CONSTANT COEFFICIENT RESONANCE ANALYSIS')
xlabel('Rotor speed ratio')
ylabel('Modal damping')
subplot(2,1,2); plot(rpm,abs(imag(eigenval(1:2:12,:)))*60/2/pi/rpmmx,'k. ');hold on; plot(rpm,rpm,'k. ');
plot(rpm,hub,'r');%plot(rpm,rpm-sqrt(nusq),'b. ');
xlabel('Rotor speed ratio')
ylabel('Modal frequency ratio')
%title('one damper inoperative')

```

## LIST OF REFERENCES

1. Cuesta, J. D., *Modeling Helicopter Blade Dynamics Using a Modified Myklestad-Prohl Transfer Matrix Method*, Naval Postgraduate School, 1994.
2. Deutsch, M. L., *Ground Vibrations of Helicopters*, Journal of Aeronautical Sciences, 1946.
3. Gerstenberger, W. and Wood, E. R., *Analysis of Helicopter Aeroelastic Characteristics in High-Speed Flight*, AIAA Journal, 1963.
4. Hammond, C.E., *An Application of Floquet Theory to Prediction of Mechanical Instability*, Journal of the American Helicopter Society, 1974.
5. Hoa, S. V., *Vibration of a Rotating Beam with Tip Mass*, Journal of Sound and Vibration, 1979.
6. Hodges, D. H., *Vibration and Response of Nonuniform Rotating Beams with Discontinuities*, Journal of the American Helicopter Society, 1979.
7. Hodges, D. H. and Rutkowski, M. J., *Free-Vibration Analysis of Rotating Beams by a Variable-Order Finite-Element Method*, AIAA Journal, 1981.
8. Johnson, W., *Helicopter Theory*, Dover, 1980.
9. Meirovitch, L., *Elements of Vibration Analysis*, McGraw-Hill Inc., 1986.
10. Peters, D. A. and Hohenemser, K. H., *Application of the Floquet Transition Matrix to Problems of Lifting Rotor Stability*, Journal of the American Helicopter Society, 1971.
11. Rutkowski, M. J., *The Vibration Characteristics of a Coupled Helicopter Rotor-Fuselage by a Finite Element Analysis*, NASA Technical Paper 2118, Ames Research Center, 1983.





## INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center.....2  
Cameron Station  
Alexandria, Virginia 22304-6145
2. Library, Code 52.....2  
Naval Postgraduate School  
Monterey, California 93943-5101
3. Chairman Code AA.....1  
Department of Aeronautics and Astronautics  
Naval Postgraduate School  
Monterey, California 93943-5106
4. Professor E. Roberts Wood.....4  
Department of Aeronautics and Astronautics  
Code AA/Wd  
Naval Postgraduate School  
Monterey, California 93943-5106
5. Professor Jon D. Raggett.....2  
Department of Mechanical Engineering  
Code ME/Ra  
Naval Postgraduate School  
Monterey, California 93943-5108
6. LT Daniel S. Hiatt.....2  
1131 Leahy Road  
Monterey, California 93940